Kyopal

PCI bus対応

パソコンNC開発キット

NC MATE-4KⅡ

ver.9.20a

ユーザーズマニュアル

株式会社キョーパル

はじめに

このたびは、パソコンNC開発キットをご購入いただきまして、誠にありがとうございます。
本マニュアルは、パソコンNC開発キット『NC MATE-4K Ⅱ』を正しく、安全にご使用いただく為の説明書です。本製品の機能を充分に活かしていただく為に、必ずご使用の前にお読みいただきますよう、お願い致します。

必ずお読みください

・本マニュアルに記載されている、安全に関する指示事項を必ずお守りください。
・当社製品のご使用にあたりましては、人身事故、火災事故、社会的な損害を生じさせないよう、お客様の責任において、機器の安全設計を行ってください。
・本製品は人命にかかわるような状況下で使用される機器への使用を意図しておりません。航空・宇宙機器、原子力制御機器、生命維持のための医療機器など、特殊用途への使用をご検討の場合は、当社までご相談ください。
・本製品を日本国外へ持ち出す場合には、外国為替及び外国貿易法の規定に従った手続きを行ってください。
・本資料の記載内容は、製品改良のため予告無しに変更する場合があります。
・当社の承諾なしに、この資料の転載・複製を禁止します。

安全にご使用いただくために

本マニュアルでは、人への危害や機器への損害をさけるため、下記の記号で安全に関する情報を提供しております。

記号	意味
⚠ 危険	この表示の記載内容を無視して、誤った取り扱いをすると、人が死亡または重症を負う危険が、切迫して生じることが想定される内容を示しています。
⚠ 注意	この表示の記載内容を無視して、誤った取り扱いをすると、人が障害を負ったり、物的損害の発生が想定される内容を示しています。

尚、[注意]に記載した事項でも状況によっては重大な結果に結びつく可能性もあります。
いずれも重要な内容を記載しておりますので、必ず守ってください。

■取扱上の注意事項

危険

・周囲に鉄粉や塵埃、可燃性および腐食性のガスがある場所で使用しないでください。
爆発、火災、感電、故障により、大けがや死亡につながります。

注意

・本製品を改造しないでください。。改造したものに対しては、当社は一切の責任を負いません。
・ボードに衝撃を与えたり、曲げたりしないでください。
・ボードの金メッキ端子部には、手を触れないでください。
・パソコンの電源が入った状態で、ボードを拡張スロットに実装したり、抜いたりしないでください。
上記のような使用をされた場合、誤作動、発熱、故障の原因になります。

■安全対策としての留意点

・システムを製造、運転する場合、取扱説明書をよく読み、安全作業を心がけてください。

・危険な個所には安全カバーを設け、そのカバーを開けた時や，取り外す時に機械が停止するようにしてください。

・運転中に作業者が危険領域に入ると、自動的に運転停止になるように安全対策を行ってください。

・事故や危険回避のため、予見できる誤操作や保守点検を考慮に入れ、警告表示を付けてください。

・不完全な加工プログラムや接触不良，誤操作，ノイズなどにより予期せぬ動作をした場合でも、安全側になるようにインターロックを充実させてください。

・鉄粉や塵埃，腐食性ガスなどが多い場所，危険を伴う用途にはご使用を避けてください。

・システムの配線は接触不良やショートなどが無いように注意して配線し、ノイズから保護するため、サージキラーやシールド処理を行ってください。

・誤作(ごさく)防止のため、原点マークを機械の適所に取り付けて、作業開始のとき正規位置にあることが確認できるようにしてください。

・システムのメンテナンスは、必ず動力電源を切って行うよう警告表示を付けてください。

マニュアルの構成

本マニュアルは、下記のような内容で構成されております。

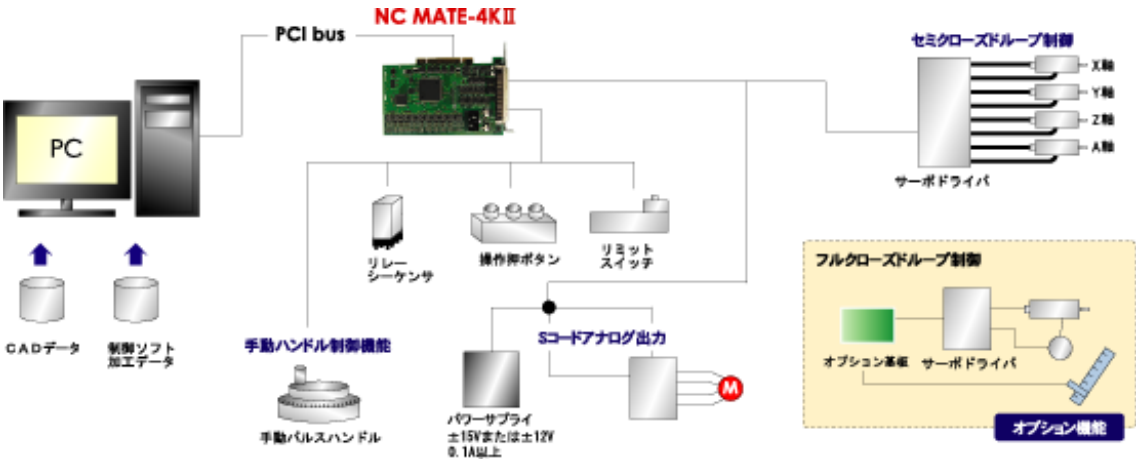
第 1 章	導入編	パソコン NC 開発キットの概要や、基板の実装方法、NC 制御ソフトのインストール方法などについて説明しています。
第 2 章	操作編	NC 制御ソフトの操作画面を使った、各種操作方法について説明しています。
第 3 章	ハードウェア編	パソコン NC 開発キットのブロック図、外形図、入出力信号、および接続例について説明しています。
第 4 章	プログラミング編	プログラムの概要、構成、G コード他各種命令および、プログラム作成要領などについて説明しています。
第 5 章	機能編	各種機能設定について説明しています。
第 6 章	応用編	パソコンNC開発キットの各種機能を使った応用的な使い方や、プログラム開発例などを掲載しています。
第 7 章	困ったときには	動作不具合が発生した場合のチェックポイントについて説明しています。
	参考資料	各種サーボモータとの接続例や、各種モータ選定条件、制御操作版参考外形図などを掲載しています。

第１章　導入編

1・1　パソコン NC 開発キットの概要

パソコン NC 開発キット NC MATE-4K Ⅱは、パソコンやサーボモータ等と組み合わせることにより、簡単な位置決めから G コードを使用した本格的な数値制御までを容易に構築することが可能です。

NC システム構成例



1・1・1　パソコン NC 開発キット BASIC の特長

パソコン NC 開発キット BASIC(NC MATE-4K Ⅱ B) は、NC 制御システムに必要な基本機能を装備しています。

・手動ハンドル制御機能　・イグザクトストップ機能　・外付け押ボタン操作機能　・スケーリング機能
・プログラマブルミラーイメージ機能　・座標回転機能　・ファイル連結機能　・S コードアナログ出力機能
・制御軸拡張機能　・クッション機能

1・1・2　パソコン NC 開発キット MASTER の特長

パソコン NC 開発キット MASTER(NC MATE-4K Ⅱ M) は、より複雑な、より高度な制御も実現します。
※ BASIC の機能に下記の機能が追加されます。
・手動介入機能　・工具補正機能　・スキップ機能　・遠隔制御運転機能　・ティーチング機能
・ヘリカル補間機能　・ピッチ誤差補正機能　・ユーザーマクロリンク機能

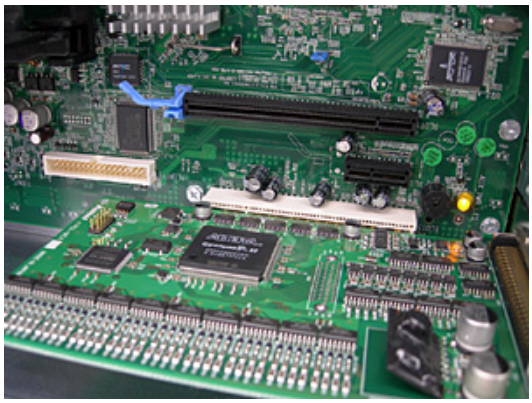
1・2　仕様一覧

仕様		内容
同時制御軸数		4 軸
補間指令		直線補間、多象限円弧補間
NC 指令		ワードアドレス　フォーマット
最大ストローク		± 9999999.999mm
最大指令値		9999999.999mm
最大送り速度		47m/ 分
オーバーライド		10 → 200%（＊外付け押しボタン操作の場合は、10 → 150%）
入力ポート		40 ポート
出力ポート		16 ポート
外部入出力用電源	定格電圧	24V
	容量	0.5A
	出力安定度	± 10%以下
	リプルノイズ	300mVp-p 以下
	コネクタへの供給	+24V (CN1 ピン番号 1，49)
登録プログラム		9999 本
最小設定単位		0.001mm
ステップ数		399999 ステップ
最大文字数		60 文字 / ステップ
使用可能 G コード*i)		G00,01,02,03,04,09,10,17,18,19,27,28,28.1,28.2,28.3,28.4,31,31.1,40,41,42,43,44,49,50,50.1,51,51.1,53,61,64,68,69,90,91,92,104,128,132,133,193,200,300,400
サブプログラム		多重ネスティング可能（実行前にサブプログラム展開する方式）
補助機能（M）		8 指令または 250 指令（動作確認を行う補助機能方式）
工具機能（T）		15 指令、M コードと併用で拡張が可能（15 ＊ 250 指令）
主軸機能（S）		± 15V または± 12V 電源入力で S コードアナログ出力が可能
手動ハンドル		手動パルスハンドル追加で、同時 2 軸のハンドル送りが可能
バックラッシュ		補正量（0 ～ 255 パルス）
電子ギア		ギア比（1/10 ～ 10）、データ（1 ～ 65535）
実行速度		約 100 ステップ / 秒

＊i)NC MATE-4K Ⅱ BASIC では、一部使用できない指令があります。

1・3　セットアップについて

1・3・1　パソコンへの設置方法

順序	説明
①	パソコン本体の電源が "OFF" であることを確認し、電源ケーブルを AC コンセントから外します。その後、外装カバーを外します。
②	 PCI スロットへ NC 制御基板を差込みます。 この時、NC 制御基板の金メッキ端子が隠れるまで差込ます。 **注意** NC 制御基板は正しい向きに実装してください。間違った向きに実装した場合、パソコン、NC 制御基板を破損する恐れがあります。
③	本体へ外装カバーをもと通り取り付け、電源ケーブルを AC コンセントに接続します。その後、パソコンを起動します。
④	以上で NC 制御基板の設置は完了です。次にドライバのインストールを行います。

注意　格納サイズの狭い PCI スロットは Low Profile 専用と呼ばれていますが、このタイプのパソコンは使用できません。

１・３・２　ドライバのインストール

１・３・２-１　WindowsXP をご使用の場合

順序	説明
①	Windows XP が起動すると「新しいハードウェアの検出ウィザード」が表示されます。
②	「一覧または特定の場所からインストールする（詳細）」が選択されていることを確認し、「次へ」をクリックします。
③	「次の場所で最適のドライバを検索する」を選択し、「次の場所を含める」にチェックをつけ、NC MATE-4K Ⅱの CD-ROM を指定し、「次へ」をクリックします。
④	ファイルのコピーが始まります。
⑤	インストールが完了した旨のダイアログボックスが表示されるので、「完了」ボタンをクリックします。
⑥	以上でドライバのインストールは完了です。 ドライバのインストールが完了したら、ソフトウェアのインストールを行ってください。

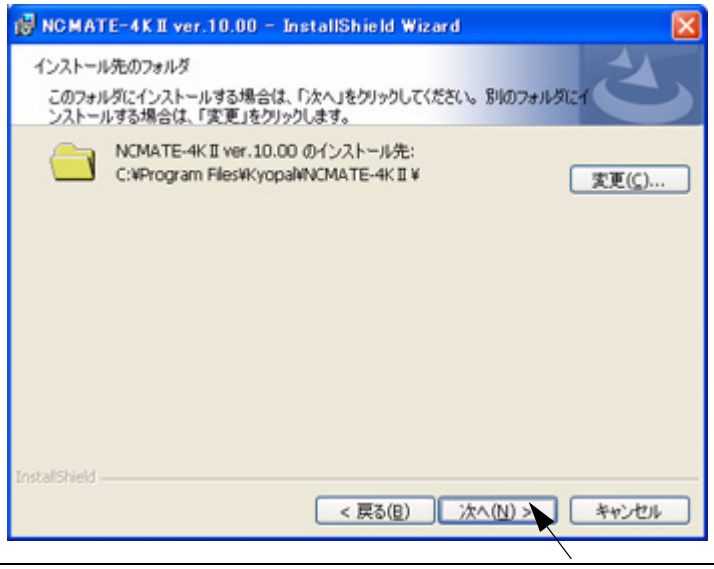
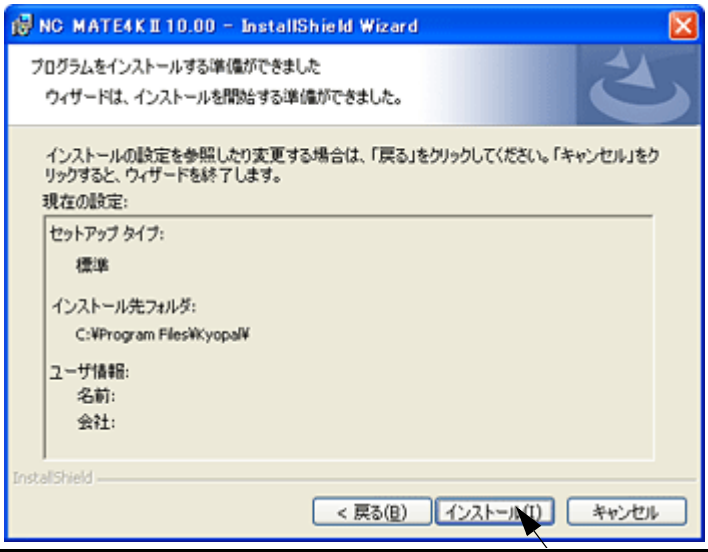
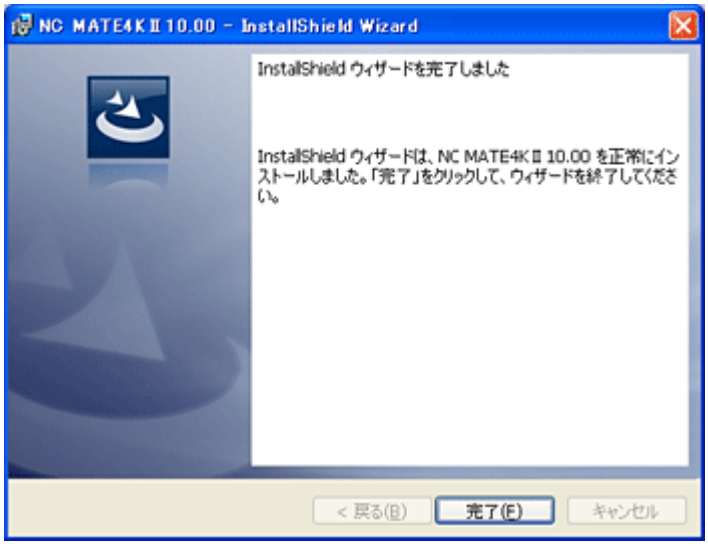
１・３・２-２　Windows 2000 をご使用の場合

順序	説明
①	Windows 2000 が起動すると「新しいハードウェアの検出ウィザード」が表示されるので、「次へ」をクリックします。
②	「デバイスドライバに最適なドライバを検索する（推奨）」を選択し、「次へ」をクリックします。
③	「検索場所のオプション」で「場所を指定」を選択し、「次へ」をクリックします。
④	「参照」ボタンをクリックし、「製造元のファイルのコピー元」に NC MATE-4K Ⅱの CD-ROM を指定します。その後、「OK」ボタンをクリックします。
⑤	ドライバが見つかった旨のダイアログボックスが表示されるので、「次へ」をクリックします。
⑥	ファイルのコピーが始まります。
⑦	以上で、ドライバのインストールは完了です。 ドライバのインストールが完了したら、ソフトウェアのインストールを行ってください。

1・4　NC 制御ソフトについて

1・4・1　NC 制御ソフトのインストール方法

手順	内容	
①	パソコンに NC MATE の CD-ROM を挿入すると、メニュー画面が立ち上がります。 ※メニューが自動的に立ち上がらない場合は、CD-ROM の中の [menu] フォルダを開き、setup_guide.exe をダブルクリックすることにより、メニュー画面が立ち上がります。	
②	次に、インストールするバージョンを選択します。 カスタマイズ可能な新しいインターフェースをご希望の場合は、[ver.10.00] をクリックします。 従来のインターフェースをご希望の場合は、[9.20a] をクリックします。	
③	NC 制御ソフトインストールウィザード*i) が起動します。 「次へ」をクリックします。	

手順	内容	
④	インストール先のフォルダを確認し、「次へ」をクリックします。	
⑤	「インストール」をクリックすると、インストールを開始します。	
⑥	インストールが完了したら、「完了」をクリックします。 これで NC 制御ソフトとユーザーズマニュアルのインストールは完了です。	

*i）この解説のインストール画面は、Windows XP のものです。

１・４・２　NC 制御ソフトの起動方法

NC 制御ソフトのインストールが完了すると、次の手順で起動させることができます。

１・４・2-1　パソコンに NC MATE-4K II 基板のセットアップが完了している場合

インストールが完了すると、デスクトップに「NCMATE4K Ⅱ Master9.20a」のアイコンが作成されます。このアイコンをダブルクリックすると起動できます。

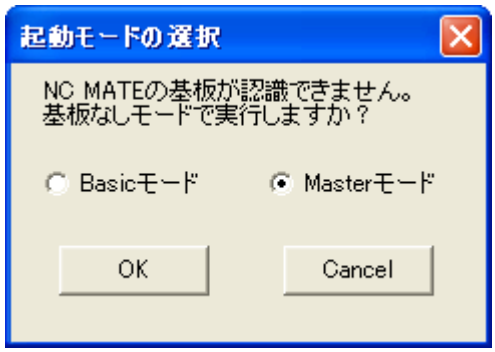
またスタートメニューからも起動できます。

スタート >> プログラム >> kyopal >> NC MATE4K Ⅱ 9.20a >> NC MATE4K Ⅱ 9.20a

をクリックすると起動できます。

⚠	ソフトウェアを、二重起動しないように注意してください。

１・４・2-2　NC MATE-4K II 基板がない場合

手順	説明	
①	デスクトップの「NCMATE4K Ⅱ 9.20a」アイコンをダブルクリックします	
②	「基板なしモードで実行しますか？」のコメントが出ますので、Basic モードか Master モードのどちらかを選択し、「OK」をクリックします。	
③	基板なしで NC 制御ソフトが立ち上がります。	

⚠	この場合、パルス列などの出力はできませんが、操作などは殆ど変わりなく実行可能です。

1・4・3　NC 制御ソフトをハードディスクに取込んだ場合の構成

ディレクトリー構成をツリー状に表示すると下記のようになります。
(エクスプローラで C ドライブの全てのファイルと拡張子を表示)

C:¥(ハードディスク)

```
Program Files
└Kyopal
  └<NC MATE-4KⅡ>
    ├<NC MATE-4KⅡ ver.9.20a>
    │ ├NcMate4K2M.exe ・・・・・・・・・・ NC 制御ソフトの実行ファイル
    │ ├kypldrvdll.dll ・・・・・・・・・・・ dll ファイル
    │ ├ncmatelib.dll ・・・・・・・・・・・・ dll ファイル
    │ ├ParaW.ncm ・・・・・・・・・・・・・・ 速度設定などパラメータのデータファイル(テキスト型)
    │ ├T_offW.ncm ・・・・・・・・・・・・・・ 工具長，工具径オフセットのデータファイル(テキスト型)
    │ ├PitchW.ncm ・・・・・・・・・・・・・・ ピッチ誤差補正のデータファイル（テキスト型）
    │ ├SimuW.ncm ・・・・・・・・・・・・・・・ [描画] で作成される描画用ファイル(テキスト型)
    │ ├ConvData.txt ・・・・・・・・・・・・・ [読込] で作成される径補正など変換されたファイル
    │ ├Data. txt ・・・・・・・・・・・・・・・ ティーチングや G132，G133 で作成されるファイル
    │ ├O1. txt
    │ │ ～                                    ・・・・・・・・・・・・・・ NC プログラム(Gコード)データファイル(テキスト型)
    │ ├O7. txt
    │ ├新規プログラム作成 txt ・・・・ 新規に NC プログラムを作成する際のファイル
    │ ├String.txt ・・・・・・・・・・・・・・ タイトルの文字列ファイル（テキスト型）
    │ ├<IMAGE>・・・・・・・・・・・・・・・・・ 画像ファイル格納ディレクトリ
    │ └<DOC>・・・・・・・・・・・・・・・・・・ ユーザーズマニュアル格納ディレクトリ
```

1・4・4　パソコンの設定について

■パソコンの仕様は下記相当を使用してください。

実行環境	パソコンの仕様
Windows Vista	CPU2GHz，メモリ 1G 以上
Windows XP	CPU1.3GHz，メモリ 512MB 以上
Windows 2000	CPU700MHz，メモリ 256MB 以上

■パソコン設定は、下記のように設定してください。

設定箇所	内容
画面設定	1024 × 768 ピクセル推奨
フォントサイズ	小さいフォント（解像度＝ 96)、または 大きいフォント（解像度＝ 120）推奨
電源設定	常にオン
スクリーンセーバー	なし
システムスタンバイ	
モニタの電源を切る	
ハードディスクの電源を切る	

第２章　操作編

２・１　各種運転方法

NC 制御ソフトによる運転方法には、下記の 3 つの方法があります。

操作方法	内容
単独運転	手動動作による運転
MDI 運転	手動データ入力による自動運転
自動運転	メモリにデータを入力することによる自動運転

２・１・１　単独運転操作方法

単独運転では、次の操作ができます。(P.16 「NC 制御ソフト操作画面」①参照)

■**手動送り**：外付け押ボタン操作、手動パルスハンドルまたはマウスによる軸の移動

操作方法	内容
マウスによる操作	モード選択の「早送り」「粗送り」「細送り」「微送り」をマウスで選択し、送り方向「+」または「-」をクリックすることにより、軸の移動ができます。
外付け押ボタンによる操作	外付けのモード選択スイッチと、オーバーライドスイッチで移動速度を選択します。送り方向（「+」「-」）押しボタンに必要な配線し、押しボタンスイッチを押すことで、軸の移動ができます。
手動パルスハンドルによる操作	手動パルスハンドルの回転、またはステップ動作で軸の移動ができます。

■**手動原点復帰**：外付け押ボタン操作、またはマウスによる軸の原点への復帰

操作方法	内容
マウスまたは 外付け押ボタンによる操作	①モード選択の「原点復帰」を選択します。 ②原点復帰させたい軸を選択します。 ③送り方向ボタンで復帰方向への軸移動操作を行います。 ④座標表示が止まるまで、送り方向ボタンを押し続けてください。 ⑤ランプが赤→青に変ったら原点復帰完了です。

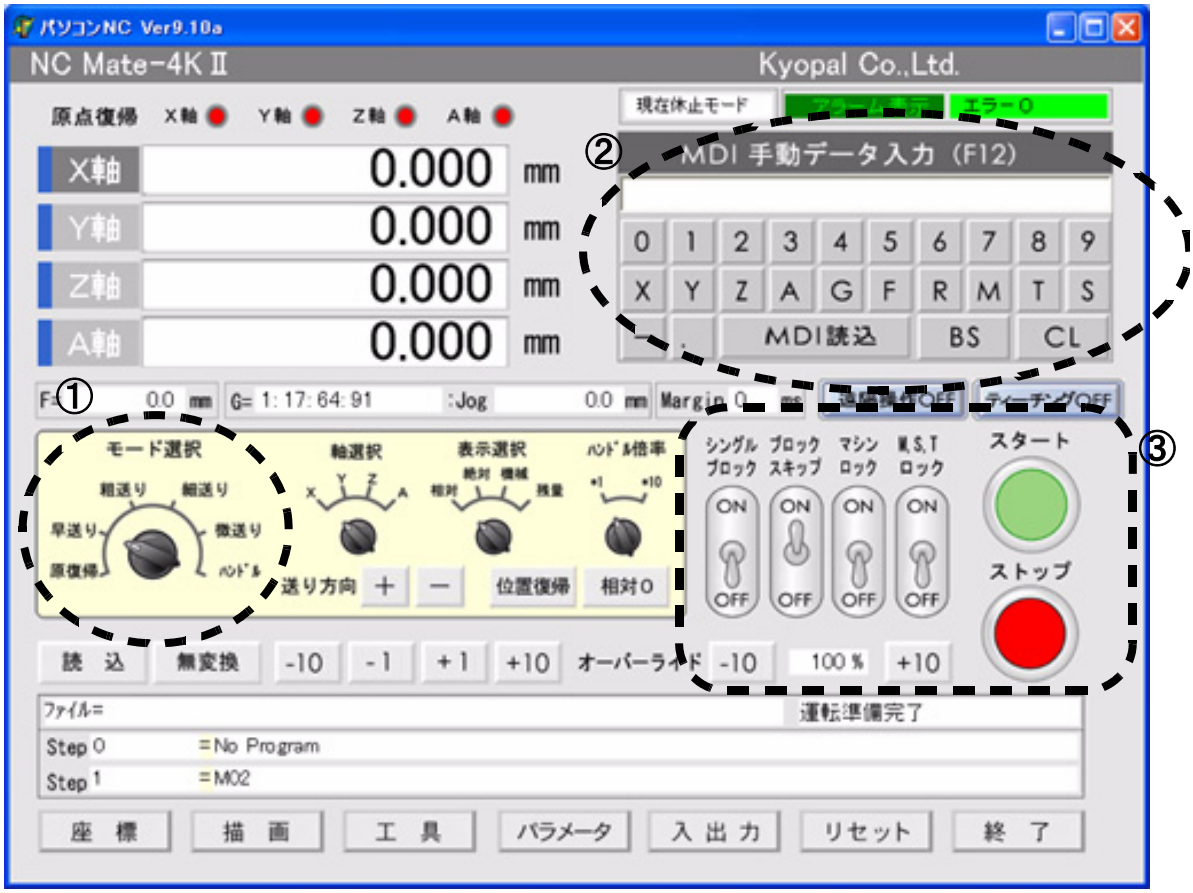
２・１・２　MDI 運転操作方法

この機能は、データの登録を必要としない簡単な指令を、マウス又はキー操作で入力し、位置決めなど、手動データ入力による自動運転を実行する機能です。

■ MDI 運転操作手順（P.16 「NC 制御ソフト操作画面」②参照）

操作方法	内容
MDI 運転による操作	キー又はマウス操作で移動データ （例：早送りで X 軸 100mm 移動させる ….G00X100）を入れます。 ↓
	MDI 読込ボタンをクリックします。 ↓
	スタートをクリックすると MDI による運転を実行します。

NC 制御ソフト操作画面*1



*1. パソコン NC ver.9__ 以下の表記は、バージョンによって異なります。

２・１・３　自動運転操作方法

自動運転は、NC プログラムを作成しメモリに入力させて操作する方法です。
NC プログラムの作成は、メモ帳やワードパッドで半角英数大文字を使用し、G コードと軸名称、移動データなどをキー入力することにより、テキスト型ファイルを作成し、記録させておきます。

■ NC プログラムの読み込み手順

「読込」をクリックすると、NC プログラム選択画面が表示されます。
データ番号（例えば 06）を選択し、「開く」をクリックしてデータを読み込み終了させます。

プログラム読込画面*1

■自動運転の操作手順（P. 16 「NC 制御ソフト操作画面」③参照）

操作方法	内容
マウスによる操作	「スタート」をクリックすると、自動運転を開始します。 ※自動運転中に「ストップ」をクリックすると、 自動運転を休止します。 ※自動運転を継続する場合は、再度「スタート」をクリックします。 ※自動運転中にオーバーライド「-10」「+10」をクリックすると、 プログラムされた送り速度を 0%～ 200%に変更できます。
外付け押ボタンによる操作	スタートボタンを押すことで、自動運転を開始します。 ※ストップボタンとオーバーライドスイッチで送り速度を 0% ～ 150 %に変更できます。

■シングルブロック　：自動運転を１ブロックごとに実行することができます。
■ブロックスキップ　：/ をつけたブロックを実行しないようにプログラムの流れを変更することができます。

*1. フォルダ名の ver.9__ 以下の表記は、バージョンによって異なります。

2・1・4　その他操作方法

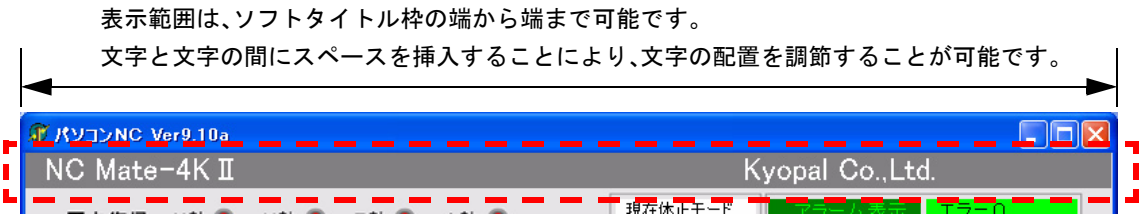
■ NC 制御ソフト操作画面の各ボタンの機能 (P.16 「NC 制御ソフト操作画面」参照)

操作ボタン名称	内容
表示選択	表示座標（相対，絶対，機械，残量）の選択ができます。
相対 0	移動量表示が全軸 0.000 となります。
「-10」「-1」 「+1」「+10」	ステップ番号を移動できます。
位置復帰	機械座標表示が全軸ゼロとなるように移動します。
座標	外部ワーク原点オフセットの値や、工具長位置オセットの現在値が表示されます。
描画	NC プログラムの読み込み後、描画画面で開始をクリックすると、軌跡や移動時間が表示できますので、プログラムのチェックや加工時間の検討が可能です。
工具[*i)]	「工具」をクリックして「工具径補正データ」または「工具長補正データ」を入力すると、G43,G44,G49 が使用できます。
パラメータ	早送り速度などの変更が可能となります。
入出力	入力信号の状態と出力の動作が確認できます。
リセット	アラームの解除とプログラムを先頭へ戻します。
終了	NC 制御ソフトが終了します。 ([f9]+[Shift]、ウィンドウ右上の [×] をクリックしても終了します)
ハンドル倍率	＊ 1 ＊ 10 は、手動ハンドルの移動単位の倍率を選択します。
無変換	工具径補正やミラーイメージ，座標回転，スケーリング，IJK → R などの変換を必要としないプログラムの高速読込が可能です。 (変換を必要とするプログラムは「読込」押ボタンを使用してください。)
マシンロック	軸移動なしで自動運転を実行します。 この機能は、主にプログラムのチェックに便利ですが、補助機能などは実行されますので注意してください。
M，S，T ロック	補助機能がロックされます。
遠隔制御[*ii)]	LAN による遠隔運転が可能です。
ティーチング[*iii)]	手動操作で軸移動を行い、「スタート」をクリックするか、または外付のスタートボタンを押すことにより、相対座標値が Data.txt に追加記録されます。このファイルをメモ帳などで編集することにより、ティーチングによるプログラム作成が容易にできます。

*i)NC MATE-4K Ⅱ BASIC では、使用できません。
*ii)NC MATE-4K Ⅱ BASIC では、使用できません。
*iii)NC MATE-4K Ⅱ BASIC では、使用できません。

■ソフトウェアタイトルの変更方法

NCMATE4K2 フォルダ内にある「String.txt」の＜タイトル＞以下の文字列を変更することにより、制御ソフト画面上部のタイトルを変更することができます。デフォルト画面では、最大全角 35 文字[*1] 設定可能です。



2・1・5　操作上の注意点

・マウスクランプ解除で使用される場合は、自動運転中に画面の移動や NC MATE-4K Ⅱ以外のソフトを操作しないでください。
・自動運転または軸移動中に常駐ソフトや他のソフトの実行，操作等はしないでください。
特にインターネットに常時接続される場合、ネットからの直接アクセスやウィルス感染等により不具合が発生する可能性がありますので、NC MATE-4K Ⅱシステム用パソコンをネットに接続される場合は注意してください。
・スキャン（パラメータ［55］）は、0=40.96，1=20.48，2=10.24，3=5.14mS で、参考値は 1=20.48 に設定されています。パラメータ値を 2 ～ 3 にしますと、1 秒間に実行できるブロック数は 100 ～ 200 ブロックとなりますが、CPU の速度は推奨仕様
Windows Vista ....CPU2GHz，メモリ 1G 以上、
Windows XP ....CPU1.3GHz，メモリ 512MB 以上、
Windows 2000..... CPU700MHz，メモリ 256MB 以上）
より速くする必要があります。
・CPU の能力判定として Margin が表示されます。この表示は常時 10ms 以上になる必要があります。この表示が 10ms 以下の場合は CPU の能力を上げるか、スキャン（パラメータ［55］）の値を少なく設定してください。

> ⚠ **注意**　Low Profile 専用タイプのパソコンは使用できません。

*1. フォントは MS P ゴシックです。制御ソフト Ver.8.0 以前の画面をご使用の場合は、最大全角 28 文字設定可能です。
＊パソコン NC ver.9__ 以下の表記は、バージョンによって異なります。

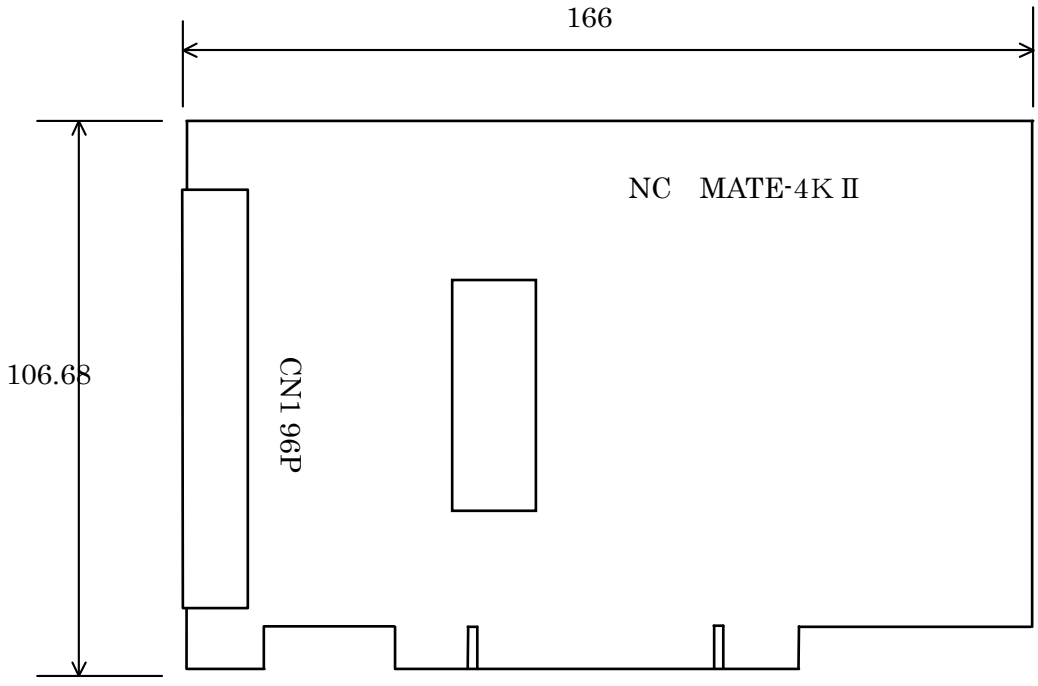
第３章　ハードウェア編

３・１　制御電源について

仕様		内容
外部入出力用電源	定格電圧	24V
	容量	0.5A
	出力安定度	± 10%以下
	リプルノイズ	300mVp-p 以下
	コネクタへの供給	+24V（CN1　ピン番号 1,49）

> **注意**　外部入出力電源は、必ずスイッチングパワーサプライなどの安定化電源をご使用ください。リップルが多い電源の場合、トラブル発生の原因になります。

３・２　NC MATE-4K Ⅱ基板寸法



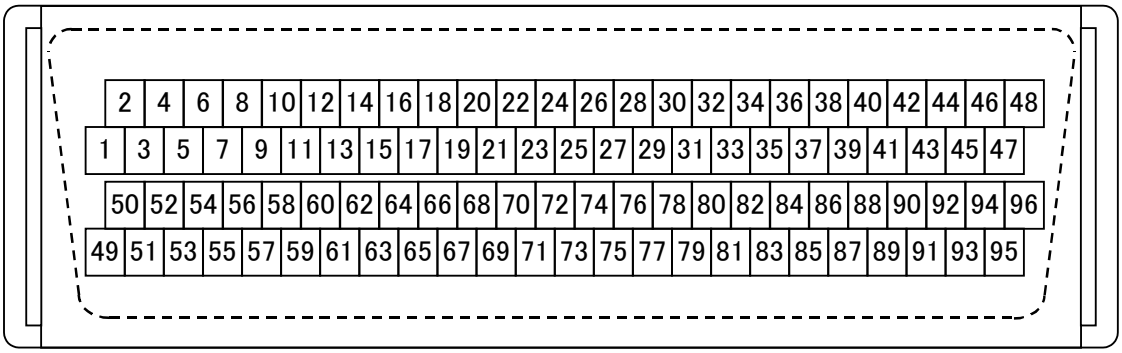
3・3　コネクタについて

3・3・1　コネクタ機種

品名	型式
CN1 用　ケーブル側コネクタ	PCR-E96FS
CN1 用　ケーブルカバー	PCS-E96LKPA（本多通信）

3・3・2　コネクタ配列図

ハンダ側から見た図



3・3・3　コネクタ接続表

No.	信号名		用途説明	No.	信号名		用途説明
48	+15V 入力	+15V	S コードアナログ出力 +15V 入力	96	-15V 入力	-15V	S コードアナログ出力 -15V 入力
47	D/A 出力	DAP	S コードアナログ出力	95	0V	0V	S コードアナログコモン
46	手動ハンドル 1A	H1A	手動ハンドル 1 信号入力 A 相	94	手動ハンドル 1B	H1B	手動ハンドル 1 信号入力 B 相
45	手動ハンドル 2A	H2A	手動ハンドル 2 信号入力 A 相	93	手動ハンドル 2B	H2B	手動ハンドル 2 信号入力 B 相
44	手動ハンドル +5V	+5V	手動ハンドル用出力 電源 5V(0.2A Max.)	92	手動ハンドル COM	COM	手動ハンドル　コモン
43	X 軸 1 回転入力 +	ZDX+	モータ 1 回転マーカ 入力信号 +	91	X 軸 1 回転入力 -	ZDX-	モータ 1 回転マーカ 入力信号 -
42	X 軸パルス出力 +	PPX+	正転、パルス列、A 相 出力 +	90	X 軸パルス出力 -	PPX-	正転、パルス列、A 相 出力 -
41	X 軸方向出力 +	PDX+	逆転、方向、B 相出力 +	89	X 軸方向出力 -	PDX-	逆転、方向、B 相出力 -
40	Y 軸 1 回転入力 +	ZDY+	モータ 1 回転マーカ 入力信号 +	88	Y 軸 1 回転入力 -	ZDY-	モータ 1 回転マーカ 入力信号 -
39	Y 軸パルス出力 +	PPY+	正転、パルス列、A 相 出力 +	87	Y 軸パルス出力 -	PPY-	正転、パルス列、A 相 出力 -
38	Y 軸方向出力 +	PDY+	逆転、方向、B 相出力 +	86	Y 軸方向出力 -	PDY-	逆転、方向、B 相出力 -
37	Z 軸 1 回転入力 +	ZDZ+	モータ 1 回転マーカ 入力信号 +	85	Z 軸 1 回転入力 -	ZDZ-	モータ 1 回転マーカ 入力信号 -
36	Z 軸パルス出力 +	PPZ+	正転、パルス列、A 相 出力 +	84	Z 軸パルス出力 -	PPZ-	正転、パルス列、A 相 出力 -
35	Z 軸方向出力 +	PDZ+	逆転、方向、B 相出力 +	83	Z 軸方向出力 -	PDZ-	逆転、方向、B 相 YYY-TYYYYYYYYYYYYYYYYYYY
34	A 軸 1 回転入力 +	ZDA+	モータ 1 回転マーカ 入力信号 +	82	A 軸 1 回転入力 -	ZDA-	モータ 1 回転マーカ 入力信号 -
33	A 軸パルス出力 +	PPA+	正転、パルス列、A 相 出力 +	81	A 軸パルス出力 -	PPA-	正転、パルス列、A 相 出力 -
32	A 軸方向出力 +	PDA+	逆転、方向、B 相出力 +	80	A 軸方向出力 -	PDA-	逆転、方向、B 相出力 -
31	自動運転中	STL	自動運転中出力 (運転中 ON)	79	自動運転休止中	SPL	自動運転休止中出力 (休止中 ON)
30	運転準備完了	READY	運転準備完了出力 (準備完了 ON)	78	読み取り信号	MF	補助機能読み込み ストローブ出力
29	出力データ 3	OUT3	出力データ 3 ビット (ON で 8)	77	出力データ 2	OUT2	出力データ 2 ビット (ON で 4)
28	出力データ 1	OUT1	出力データ 1 ビット (ON で 2)	76	出力データ 0	OUT0	出力データ 0 ビット (ON で 1)
27	M コード 7 ビット	M7	補助機能 M コード出力 (ON で 128)	75	M コード 6 ビット	M6	補助機能 M コード出力 (ON で 64)
26	M コード 5 ビット	M5	補助機能 M コード出力 (ON で 32)	74	M コード 4 ビット	M4	補助機能 M コード出力 (ON で 16)
25	M コード 3 ビット	M3	補助機能 M コード出力 (ON で 8)	73	M コード 2 ビット	M2	補助機能 M コード出力 (ON で 4)

No.	信号名		用途説明	No.	信号名		用途説明
24	Ｍコード１ビット	M1	補助機能Ｍコード出力(ONで2)	72	Ｍコード０ビット	M0	補助機能Ｍコード出力(ONで1)
23	出力コモン	COM	出力共通コモン	71	出力コモン	COM	出力共通コモン
22	Ｘ軸正転　OT	+LX	Ｘ軸＋方向オーバートラベル入力	70	Ｘ軸逆転　OT	-LX	Ｘ軸－方向オーバートラベル入力
21	Ｙ軸正転　OT	+LY	Ｙ軸＋方向オーバートラベル入力	69	Ｙ軸逆転　OT	-LY	Ｙ軸－方向オーバートラベル入力
20	Ｚ軸正転　OT	+LZ	Ｚ軸＋方向オーバートラベル入力	68	Ｚ軸逆転　OT	-LZ	Ｚ軸－方向オーバートラベル入力
19	Ａ軸正転　OT	+LA	Ａ軸＋方向オーバートラベル入力	67	Ａ軸逆転　OT	-LA	Ａ軸－方向オーバートラベル入力
18	Ｘ軸原点減速	DCX	Ｘ軸原点減速開始入力	66	Ｙ軸原点減速	DCY	Ｙ軸原点減速開始入力
17	Ｚ軸原点減速	DCZ	Ｚ軸原点減速開始入力	65	Ａ軸原点減速	DCA	Ａ軸原点減速開始入力
16	非常停止	ESP	非常停止信号入力	64	補助機能完了	FIN	補助機能読み込み完了入力
15	スキップ信号	SKP	スキップ信号入力	63	位置決め完了	POS	位置決め完了入力
14	自動運転起動	ST	自動運転起動入力	62	自動運転休止	SP	自動運転休止入力
13	シングルブロック	SBK	１ブロック実行入力	61	ブロックスキップ	BDT	１ブロック無視入力
12	オーバーライド	OV3	オーバーライド３	60	オーバーライド	OV2	オーバーライド２
11	オーバーライド	OV1	オーバーライド１	59	オーバーライド	OV0	オーバーライド０
10	送り方向正	+DIR	手動送り方向(+)入力	58	送り方向逆	-DIR	手動送り方向(-)入力
9	原点復帰選択	ZRN	手動原点復帰選択入力	57	早送り選択	RT	手動早送り選択入力
8	荒送り選択	JOG	手動荒送り選択入力	56	手パ選択	HAND	手動パルスハンドル選択入力
7	手パ×10	MP10	手動パルスハンドル倍率×10	55	位置復帰	HOME	位置復帰入力
6	Ｘ軸選択	SELX	Ｘ軸手動送り選択入力	54	Ｙ軸選択	SELY	Ｙ軸手動送り選択入力
5	Ｚ軸選択	SELZ	Ｘ軸手動送り選択入力	53	Ａ軸選択	SELA	Ａ軸手動送り選択入力
4	入力データ３	IN3	入力データ３ビット(ONで8)	52	入力データ２	IN2	入力データ２ビット(ONで4)
3	入力データ１	IN1	入力データ１ビット(ONで2)	51	入力データ０	IN0	入力データ０ビット(ONで1)
2	外部入力電源０Ｖ	0V	０Ｖ電源	50	外部入力電源０Ｖ	0V	０Ｖ電源
1	外部入力電源２４Ｖ	24V	２４Ｖ電源	49	外部入力電源２４Ｖ	24V	２４Ｖ電源

注意

① CN1 の信号名に※印がついているものは通常接点閉として使用します。

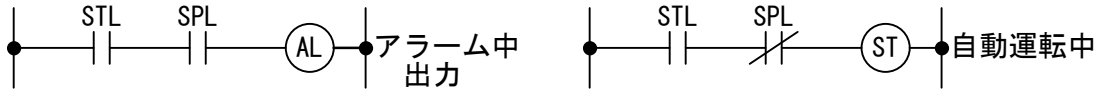
② CN1 の下記番号は必ずツイストペアーで配線を行ってください。

43-91 番，42-90 番，41-89 番 ..... X 軸関係
40-88 番，39-87 番，38-86 番 .....Y 軸関係
37-85 番，36-84 番，35-83 番 ..... Z 軸関係
34-82 番，33-81 番，32-80 番 ..... A 軸関係

ツイストペアー配線は高速パルス信号に適した方式で、ペアーの相手と互いに捩じり合うことでノイズの侵入を防ぐ効果があり、AWG28(導体部の直径が 0.32mm)程度の電線を 1cm につき 1 回ほど捩じって作成します。

③運転準備完了(READY)信号は制御ソフトが立ち上がりますと ON となり、終了させますと OFF となります。

④アラーム中には(STL)と(SPL)の出力が同時に ON となりますので、アラーム中の信号が必要な場合は合成して使用してください。



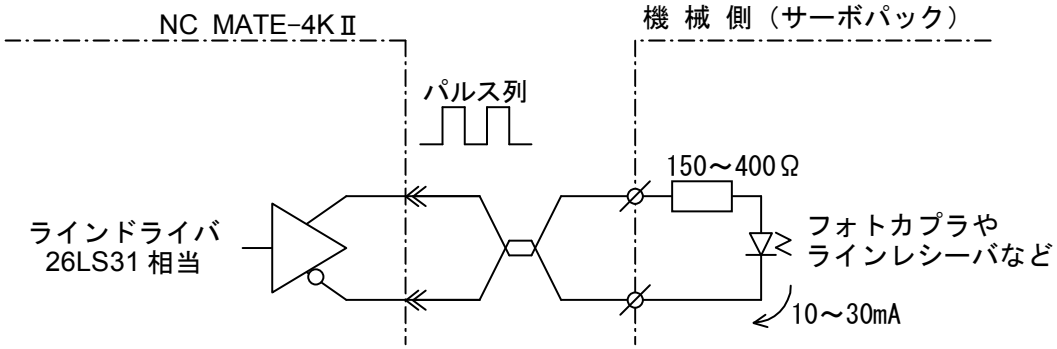
3・3・4　入出力信号規定

3・3・4-1　指令パルス出力信号(コネクタ番号 CN1)

指令パルス出力信号は、NC MATE-4K Ⅱから、機械側のサーボパック側へ送り出されるパルス列出力信号です。

●出力容量は DC 5V，40mA 以下です。

●出力はラインドライバー使用。

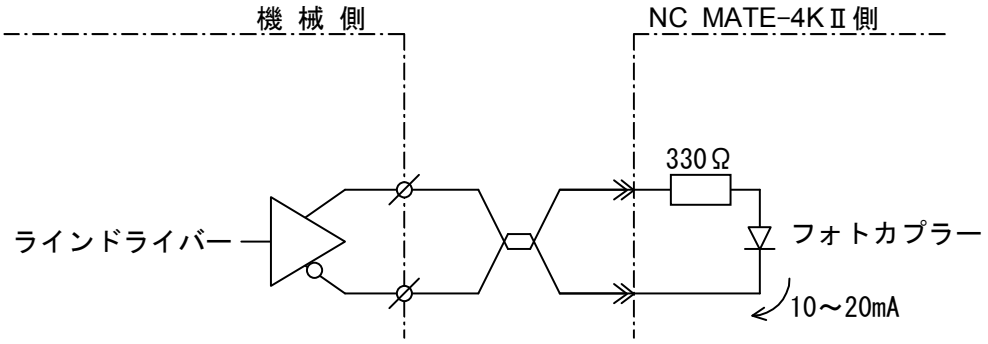


注意　指令パルスの配線は高速パルス信号で、かつ外部ノイズの影響を受け易いためツイストペアー線を使用し、強電回路とは離して最短距離（10m 以内が望ましい）で配線してください。

３・３・4-2　X, Y, Z, A 軸モータ１回転信号（コネクタ番号 CN1）

X, Y, Z, A 軸モータ 1 回転信号は、機械側のサーボパックなどから NC MATE-4K Ⅱに送られてくる信号です。

●入力電圧は MAX. 5V です。



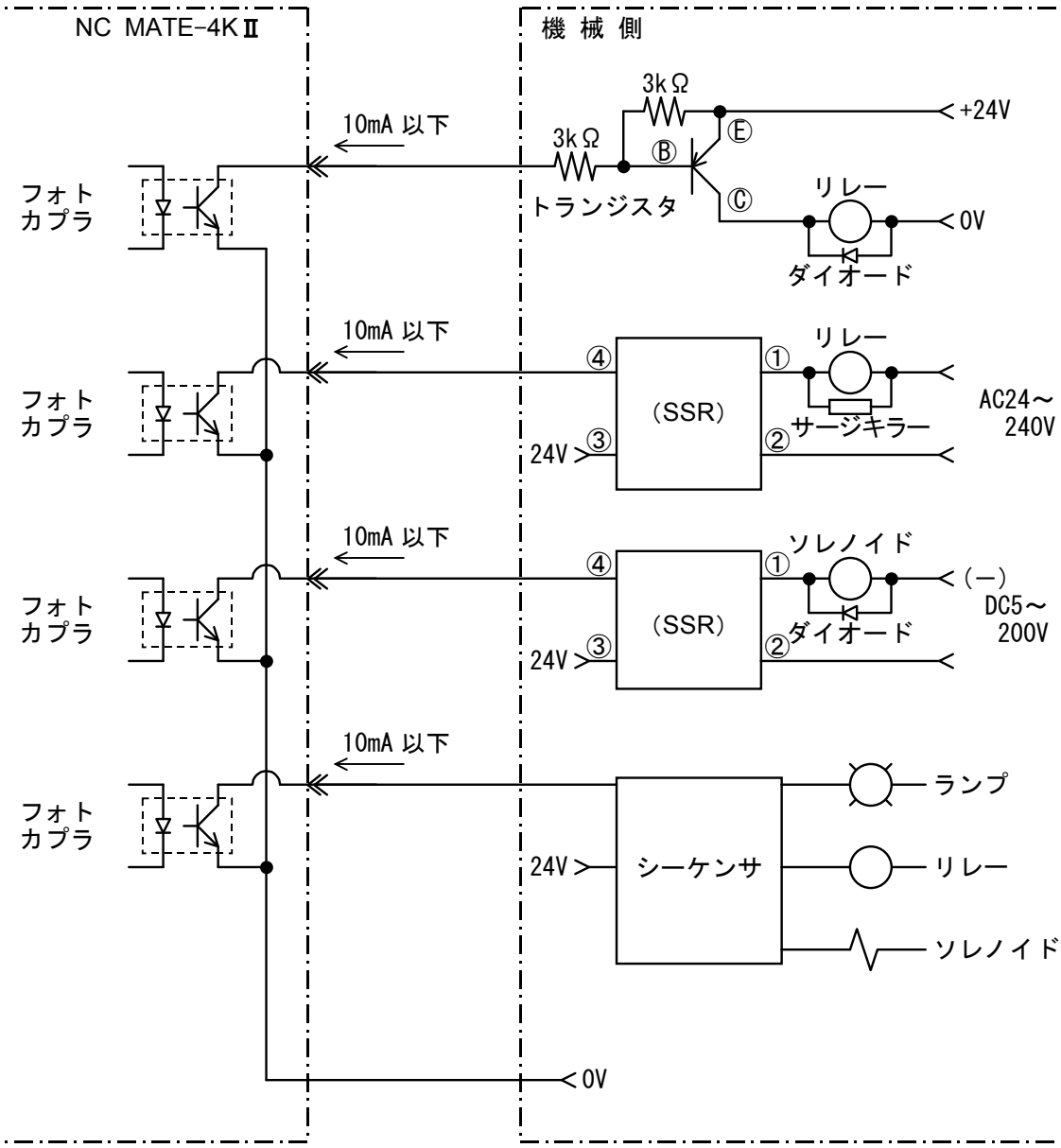
信号およびピン番号は、P. 22 「コネクタ接続表」を参照してください。

> **注意**　ラインドライバー以外（オープンコレクターなど）と接続される場合、入力電圧を 5V 以上にしますと、NC MATE-4K Ⅱ側 330Ω が焼損しますので注意してください。

3・3・4-3　外部出力信号(コネクタ番号 CN1)

外部出力信号は、NC MATE-4K Ⅱから機械側に送り出される信号です。
出力容量および保護対策は次の通りです。

●出力容量は DC 30V，10mA 以下です。定格を超えますと壊れますので注意してください。
●出力は無接点出力（フォトカプラ）です。
●無接点出力の保護として次の対策が必要です。

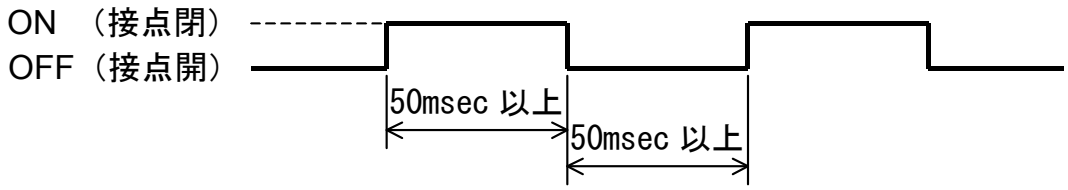


⚠ 注意	リレーやソレノイドは直接駆動することはできませんので、必ずトランジスタまたは、ソリッドステートリレー（SSR），シーケンサなどを経由してください。

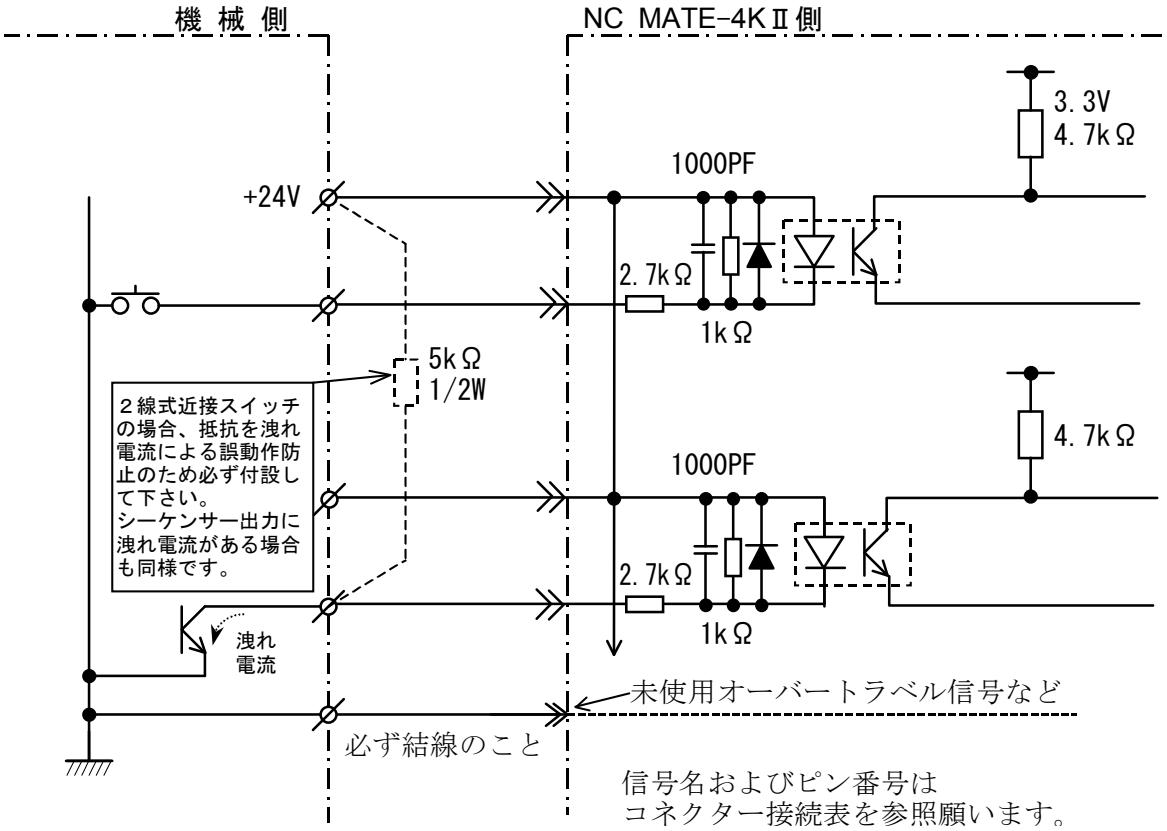
３・３・4-4　外部入力信号（コネクタ番号 CN1）

外部入力信号は、機械側から NC MATE-4K Ⅱに送られてくる信号です。
入力電圧，有効時間は次の条件です。

●入力信号は 50msec 以上の ON または OFF 信号を有効とします。



●入力接点容量は 24V，10mA とします。



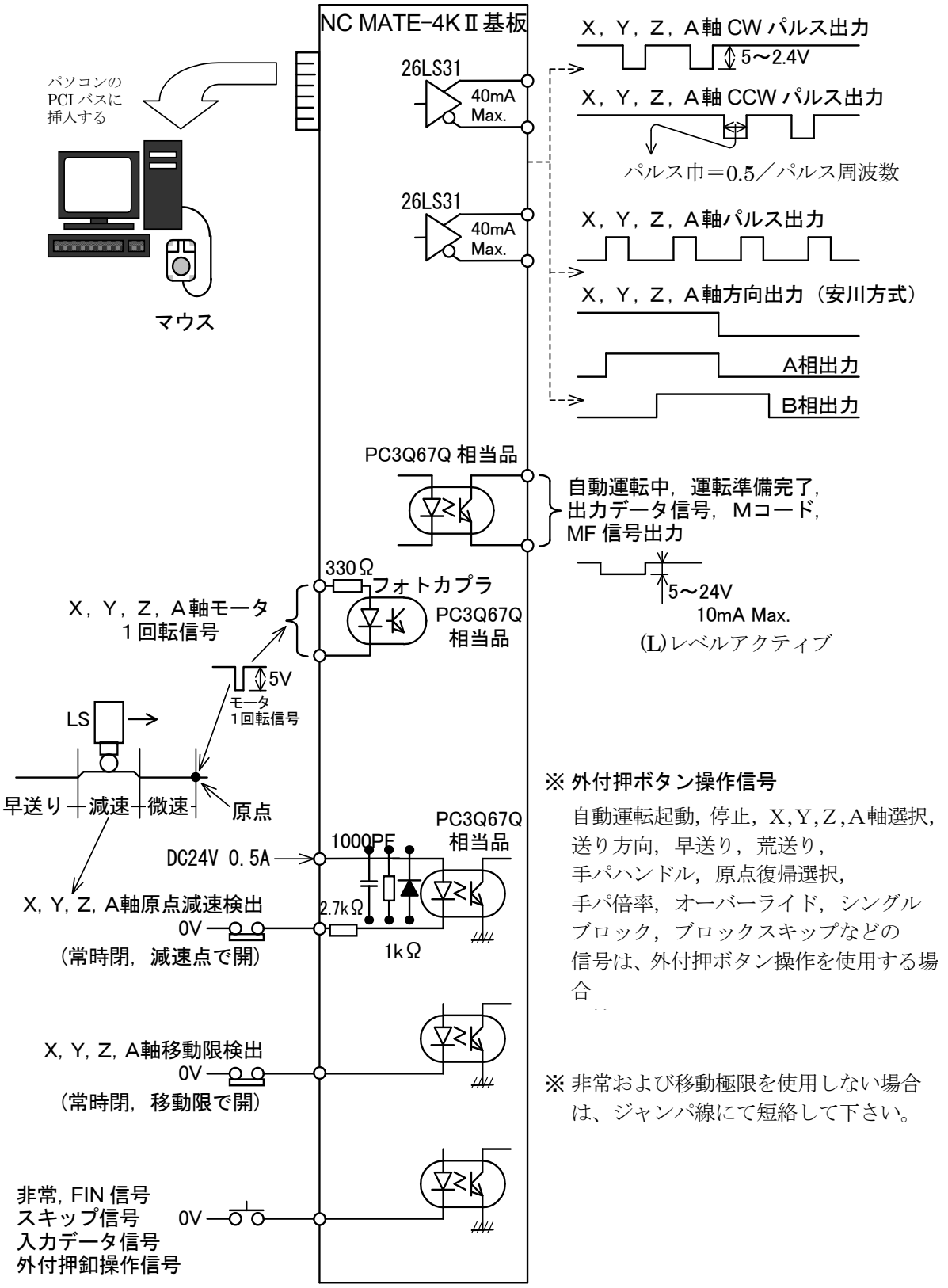
⚠ 注意	各軸オーバートラベル信号、各軸原点減速信号、非常停止、自動運転休止信号は、通常 ON(ノーマルクローズ：接点閉）として使います。 未使用の上記信号線は、接点閉となるようにジャンパで結線してください。

3・3・5　CN1 配線上の注意点

・ピン番号を間違えないように配線してください。パソコンや基板類が破損する場合があります。
・隣接するピンがショートしないように、ハンダ部を絶縁チューブで保護してください。
・ハンダ付け不良を防ぐため、線とコネクターピンはあらかじめ別々にハンダメッキを行って接合してください。
・ツイストペアー配線の指定があるピンは、ペアーの相手を間違えないように注意し、最短距離で配線してください。
・ツイストペアー配線とは、ペアーの相手と互いにねじり合って配線する方法です。
・CN1 の配線はノイズ発生の可能性がある強電回路からできるだけ離してください。
・CN1 に配線する電線のサイズは、AWG28 または導体部の直径が 0.32mm 程度の電線を推奨します。

3・4　サーボモータ接続例

NC MATE-4KⅡの接続



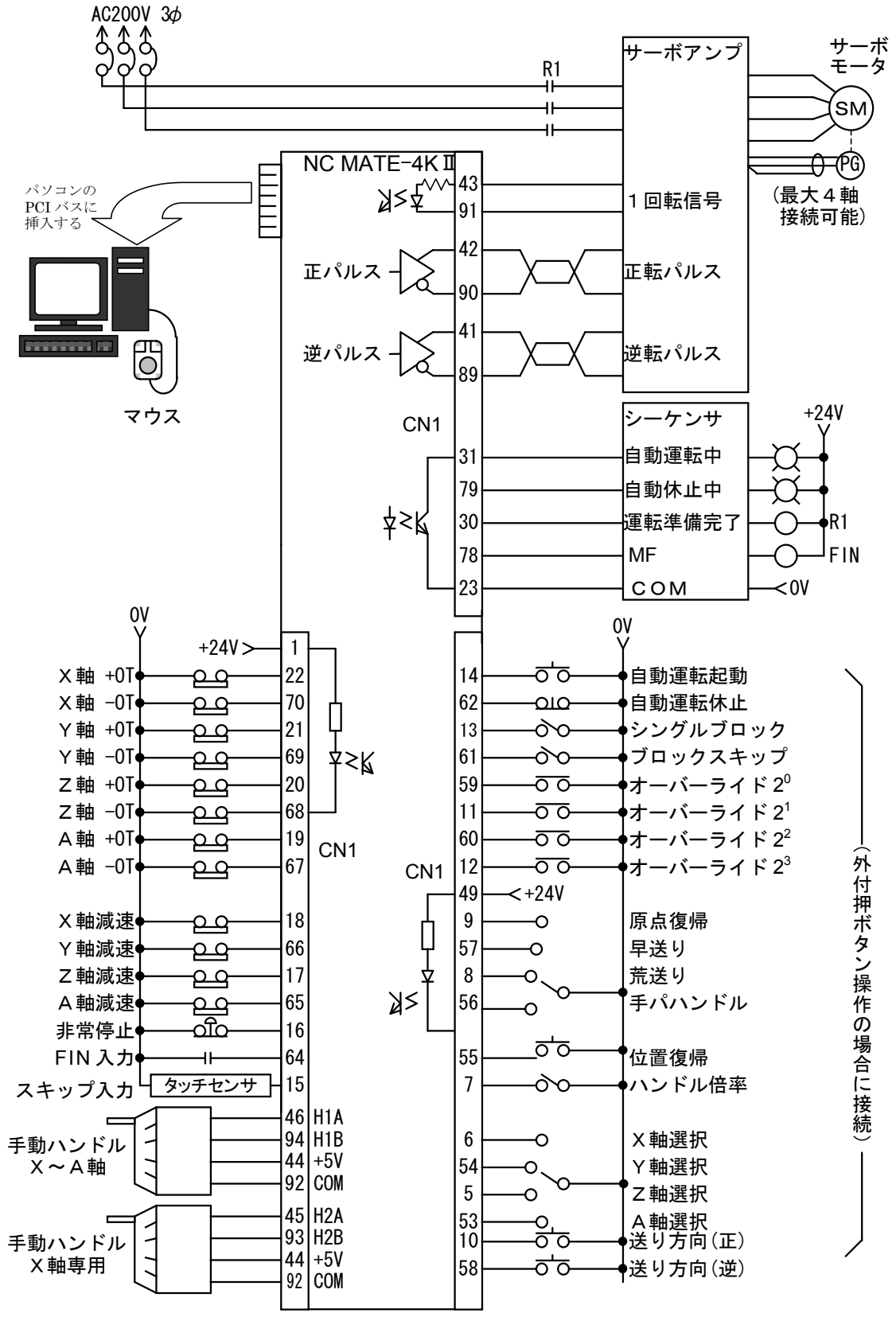
※ 外付押ボタン操作信号

自動運転起動，停止，X,Y,Z,A軸選択，送り方向，早送り，荒送り，手パハンドル，原点復帰選択，手パ倍率，オーバーライド，シングルブロック，ブロックスキップなどの信号は、外付押ボタン操作を使用する場合

※ 非常および移動極限を使用しない場合は、ジャンパ線にて短絡して下さい。

3・5　NC MATE-4K Ⅱ参考回路図

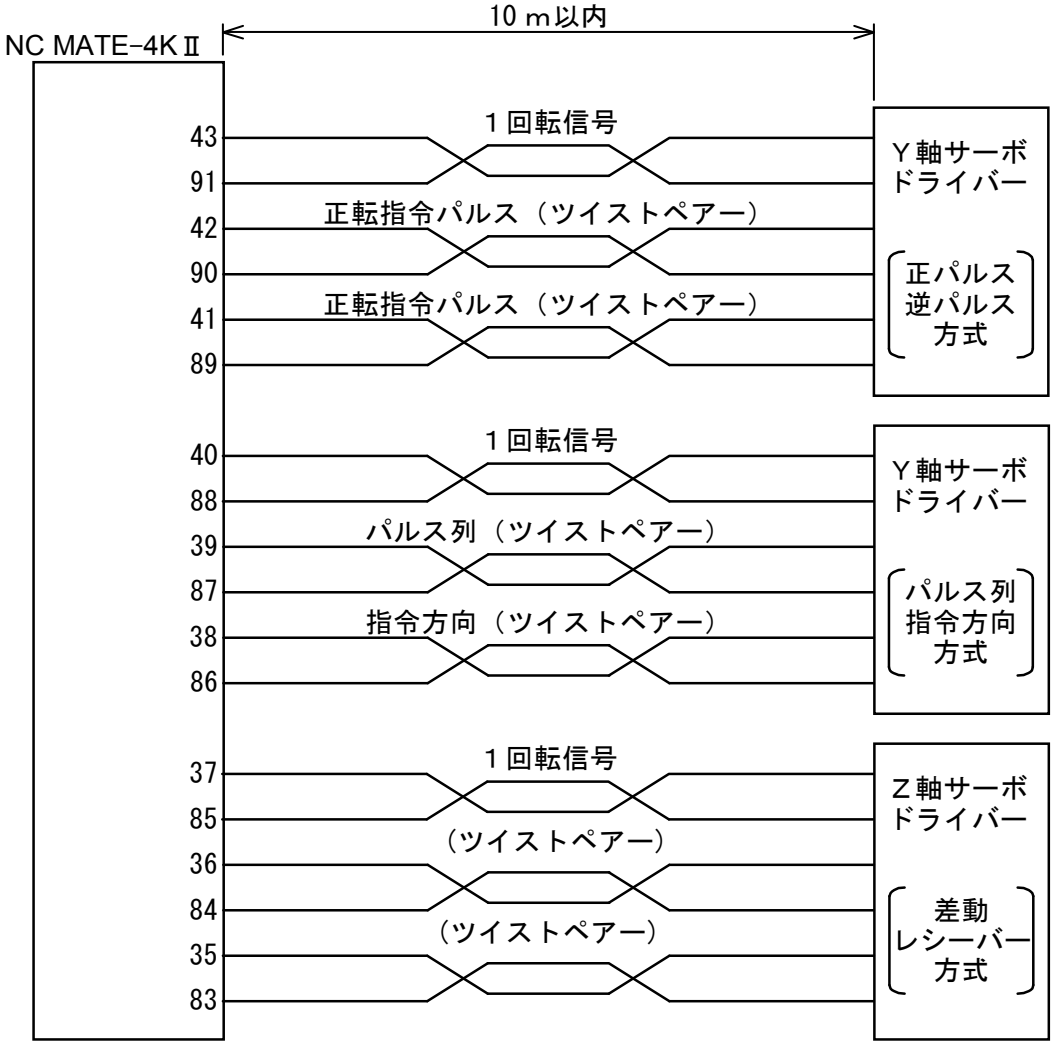
NC MATE-4KⅡ参考回路図



3・6　基板の取り付け要領

NC MATE-4K Ⅱ基板を制御盤内に取り付ける際は、下記の点に御注意ください。

・基板類やパソコンは、電磁開閉器やリレーなど、ノイズの発生源からできるだけ離して取り付け、配線も強電回路とは分離して結線してください。やむを得ず強電回路と同一ダクト内に配線する場合は、シールド付のジッパーチューブで基板関係の配線を包み、シールド側は必ず接地してください。
・指令パルスの配線は高速パルス信号のためツイストペアー線を使用し、最短距離で配線してください。
※指令パルスの配線が悪いと、位置決め精度が悪くなる場合があります。



3・7　運転上の注意点

各メーカーのサーボパックの電源投入，遮断時のシーケンスの組み方，運転前の点検，電源投入の際の注意点をチェックしてください。

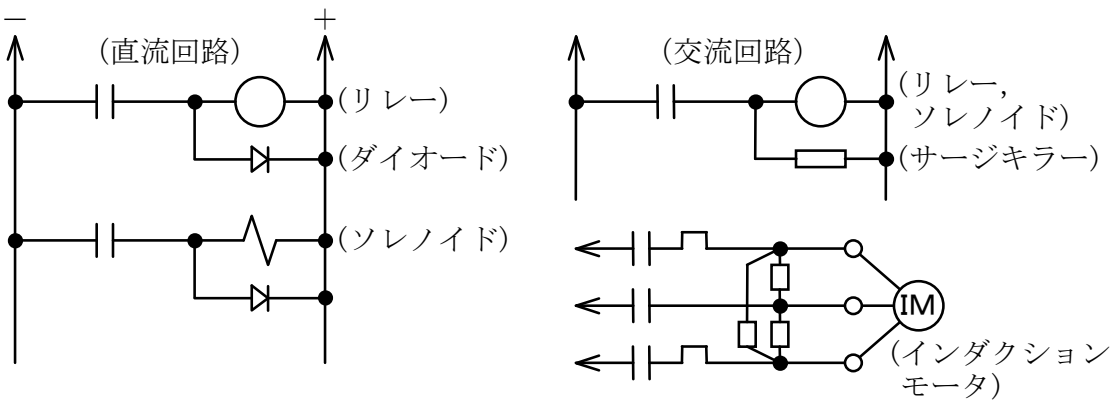
・電源ユニットの供給電圧，主回路が短絡していないかどうか、正しく配線されているかなど、チェックしてください。
・フィードバック用検出器の不具合（不良および誤配線）によるモータ（機械）の暴走には予防策がとれませんので、試運転を始めるに当たっては予期せぬ事故を避けるため、サーボモータは機械から外し、無負荷で運転して異常なしと認められた時に機械側と直結してください。

パルス当たりの移動量

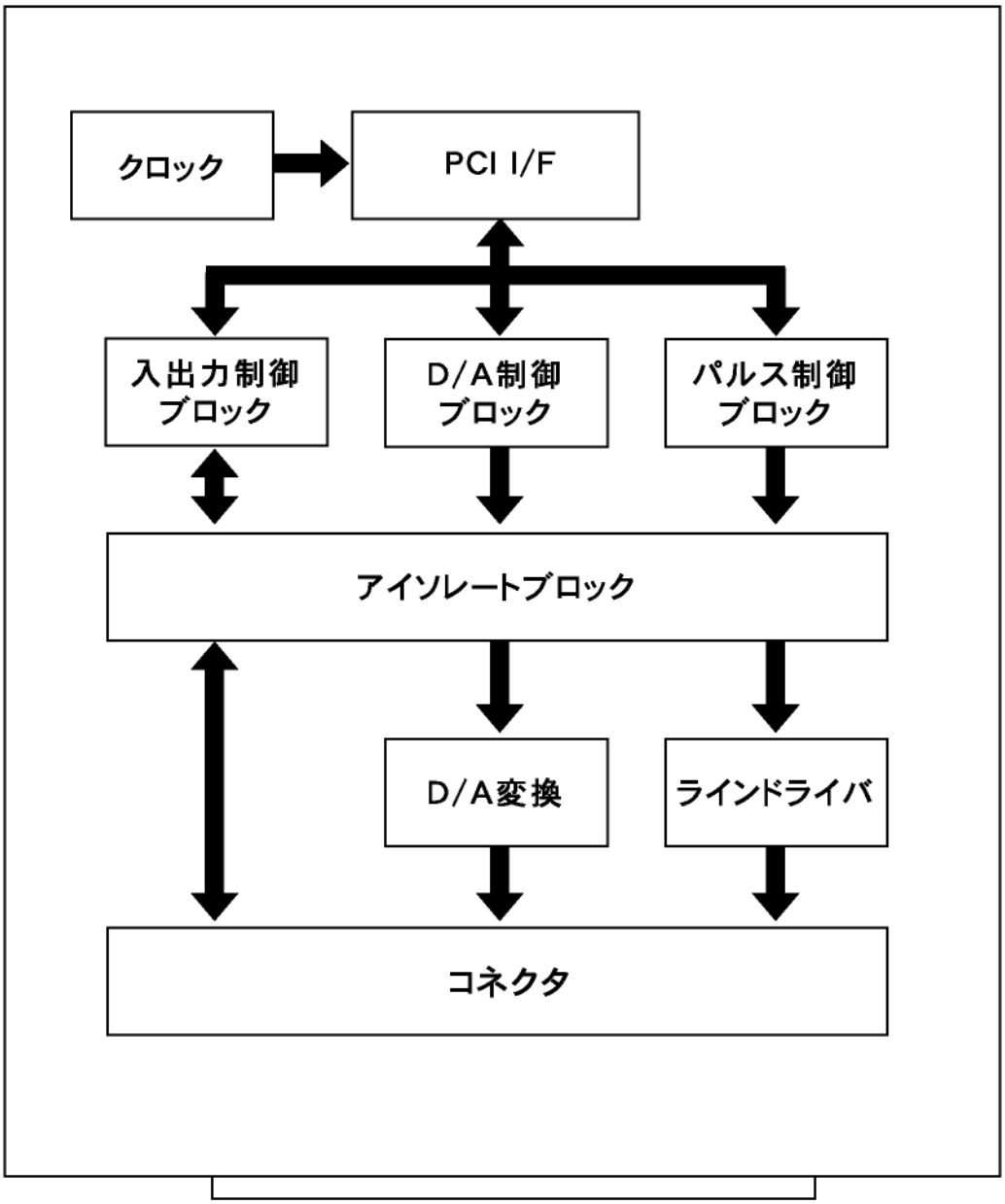
$$= \frac{\text{モータ１回転当たりの移動量}}{\text{モータ１回転当たりのフィードバックパルス数（パルスエンコーダ）}} \quad = 0.001\text{mm（deg.）}$$

上記の関係式を満足するよう、伝導系を設定してください。

・パソコンやプリント基板の破損を防ぐため、漏電個所がないかテスターで良くチェックしてください。
漏電がありますと、予期せぬ所へ電流が流れ、故障が発生する場合があります。
制御盤やパソコン本体は完全に接地し、漏電個所がないように注意してください。
・電子回路をノイズから保護するため、ノイズを発生する機器にはサージキラー，ノイズを受けやすい部分にはシールド処理を施してください。
基板類は動力回路やノイズを発生する回路とはなるべく離して配置し、配線も分離または遮蔽して結線してください。
サージキラーは、リレーやソレノイドなどのコイルに直流回路の場合は、ダイオード（DC220V 1A 以上），交流回路の場合は CR タイプ（AC 250V　0.2μF　150Ω 程度）のサージキラーを必ず取り付けてください。



3・8　NC MATE-4K Ⅱ基板ブロック図



第４章　プログラミング編

4・1　概要

NC のプログラムは、指令単位であるブロックの集合体によって構成されます。
ブロックとは、機械の動作を制御するために必要な情報の集まりで、準備機能（G コード）、補助機能（M コード）、工具機能（T コード）、主軸機能（S コード）、位置情報、送り速度などで構成されます。このブロックを機械の動作の順序に並べたものがプログラムです。

指令の集まりを《**ブロック**》と言います。

ブロックの集まりを《**プログラム**》と言います。

プログラムの最後には、必ず M02 プログラム終了コードを書きます。

・ブロックの体系

＊G_____　X_____　Y_____　F_____
準備機能　　補間機能　　送り 機能

＊M_____
補助機能

4・2　プログラムの構成

プログラムは以下に示される各部から構成されます。

- プログラム名
- タイトル
- メインプログラム（ブロック）
- エンドオブプログラム
- サブプログラム（ブロック）
- サブプログラムエンド

この中で、直接動作に関係するものはメインプログラムとサブプログラム（必要な場合）です。
その他のものは編集過程で自動的に付加されます。

プログラム名

NC MATE-4K Ⅱでは、加工のプログラムを登録できます。それらを区別するためにプログラム名称を付けます。

【例】(O21.txt)(ABC1.Txt)(ﾃｽﾄ.TXT)(**試運転**.TXT) など

タイトル

プログラム番号だけではその内容が把握できないものに簡単なタイトルを付けることができます。

【例】(**外周仕上加工プログラム**)(TEST PROGRAM) など

メインプログラム

準備機能や補助機能などの指令の集まりであるブロックの集合です。

【例】(G01 X100 Y200 F300) など、英数大文字半角で記載

エンドオブプログラム

M02 または M30 の指令ブロックでメインプログラムの終了を示します。

サブプログラム（必要な場合）

O1 ～ O9999 の番号の付いたプログラムです。

サブプログラムエンド

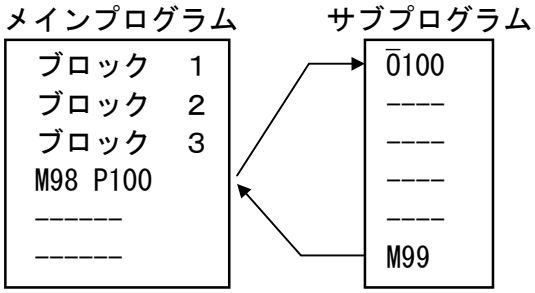
M99 の指令でサブプログラムの終了を示します。

> ⚠ **プログラムは、必ず半角の英数大文字で記載してください。**

４・２・１　メインプログラムとサブプログラム

４・２・１-1　メインプログラム

プログラムは、メインプログラムとサブプログラムに別れます。
通常、自動運転ではメインプログラムに従って動きますが、M98 コードがあると、指示されたサブプログラムを実行します。
サブプログラム実行中に M99 コードがあると、再びメインプログラムに戻ります。



４・２・１-2　サブプログラム

プログラム中で、ある固定した動作や繰り返し同じパターンが現れる場合、これをサブプログラムとして 20 本まで予めプログラム中に登録しておくことができます。

◆ **サブプログラム呼び出し**
サブプログラムを呼び出すには、M98 に続きサブプログラム番号と繰り返し回数を指令します。
L を指令しない場合は１回とみなします。

M98 P＿ L＿
P ：1 ～ 9999 （サブプログラム番号）
L ：1 ～ 999 （繰り返し回数）

◆ **サブプログラムのフォーマット**
一つのサブプログラムは、次のようなフォーマットになります。

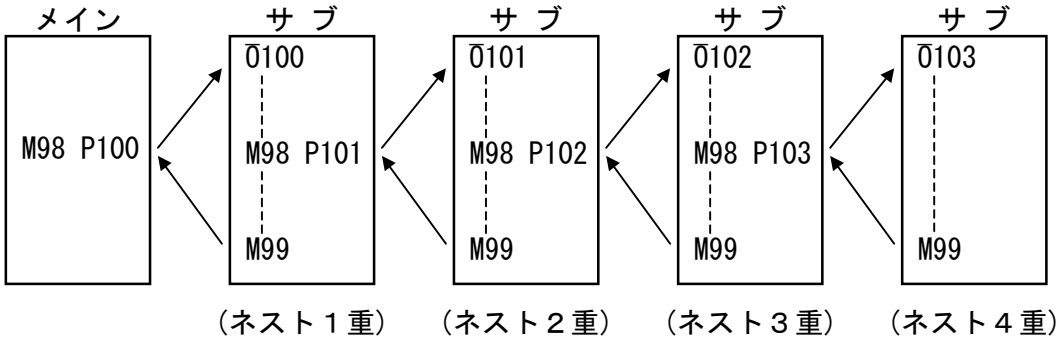
O100 → サブプログラム番号　 O1 ～ O9999

----　動作プログラム

M99　→エンドオブプログラム

◆ サブプログラムからの呼び出し

一つのサブプログラムから別のサブプログラムを呼び出すこともできます。
同様に４重のネスティングまで可能です。



・アドレスＰで指定したサブプログラム番号が見つからなかったり不適当だった場合は「サブプログラムエラー」になります。
・サブプログラムは、メインプログラムの後にプログラムされていることが必要です。
・プログラムを実行する際は、サブプログラム呼び出し時間を最小にするため、実行前にサブプログラムを展開して実行します。

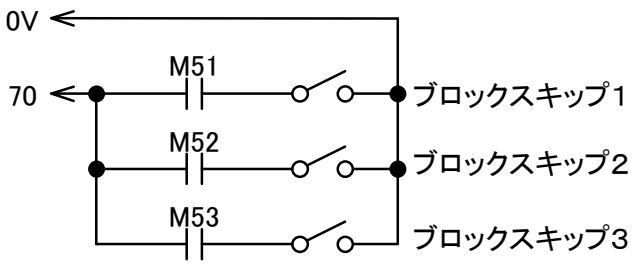
4・2・2　シングルブロック

自動運転中にシングルブロックにすると、プログラムを１ブロックずつ実行します。

4・2・3　ブロックスキップ

ブロックスキップ信号が ON の時、ブロックの先頭に “/” コードが入っているブロックを無視して読み飛ばします。（主に研削加工時の定寸に使用）
この信号を有効にしない場合は “/” コードが入っていても何ら影響はありません。
また外付押ボタン操作の場合、この信号をＭコードで切り換えることによりブロックスキップ数を拡張することが可能です。

```
M51
/G01X0.01
/G01Y100
M52
/G01X0.005
/G01Y-100
M53
```



ブロックスキップは、サブプログラム呼び出しの先頭に付けることはできません
使用不可例　/M98P100

4・3　準備機能（G コード）について

準備機能（G コード）は、G に続く数値によって、そのブロックの命令がどのような意味を持つのかを指示します。
G コードには、位置決め、補間、平面指定、リファレンス点復帰、工具補正、座標系設定、固定サイクル、アブソリュート・インクリメンタル指令などがあります。

また、G コードには次の二種類があります。

種類	内容
ワンショットの G コード	指令されたブロックのみ、その G コードが意味を持つもの
モーダルな G コード	同一グループのほかの G コードが現れるまで、その G コードが有効なもの

```
G01 X__ F__;
        Z__;  } この間は G01 モードへ
        X__;
G00 Z__;        G00 モードへ
```

4・3・1　G コード一覧表*1

G コード一覧表

コード	グループ	内容
G00	01	位置決め
G01		直線補間
G02		円弧補間 CW（時計回り）
G03		円弧補間 CCW（反時計回り）
G04	00	ドウエル（時計待ち）
G09		イグザクトストップ（ワンショット）
G10		工具オフセットデータの設定
G17	03	XY 平面指定（円弧補間）
G18		ZX 平面指定（円弧補間）
G19		YZ 平面指定（円弧補間）
G27	00	リファレンス点チェック機能
G28		リファレンス点復帰
G28.1		X 軸リファレンス点復帰（ドグ検出方式）
G28.2		Y 軸リファレンス点復帰（ドグ検出方式）
G28.3		Z 軸リファレンス点復帰（ドグ検出方式）
G28.4		A 軸リファレンス点復帰（ドグ検出方式）
G31		スキップ機能
G31.1		論理反転スキップ機能

Ｇコード一覧表

コード	グループ	内容
G40	06	工具径補正　キャンセル
G41		工具径補正　左側
G42		工具径補正　右側
G43	07	工具長補正　+
G44		工具長補正　−
G49		工具長補正キャンセル
G50	10	スケーリングキャンセル
G51		スケーリング
G50.1	11	プログラマブルミラーイメージキャンセル
G51.1		プログラマブルミラーイメージ
G53	00	機械座標系選択
G61	09	イグザクトストップモード
G64		イグザクトストップモードキャンセル
G68	12	座標回転
G69		座標回転キャンセル
G90	04	アブソリュート指令
G91		インクリメンタル指令
G92	00	ワーク座標の設定
G100		手動介入キャンセル（パラメータ 104 番で、モーダルを選択時に有効）
G104		手動介入
G128	00	任意位置原点復帰指令
G132		相対座標ファイル記録
G133		相対座標ファイルのクリア
G193		相対座標の変更
G200		ファイル連結指令
G300		遠隔制御有効指令
G400		ユーザーマクロリンク指令

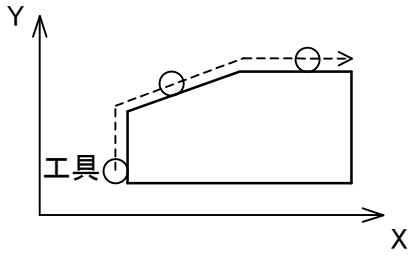
・00 グループの G コードは、ワンショットの G コードであることを示し、指定されたブロックのみで有効です。
・同一ブロックに最大 4 個の G コードを指令できます。
・同一ブロックに同じグループに属する G コードを指令しないでください。指令した場合、後で指令した G コードが有効となります。

*1. NC MATE-4K Ⅱ BASIC では使用できません。

４・３・２　補間機能について

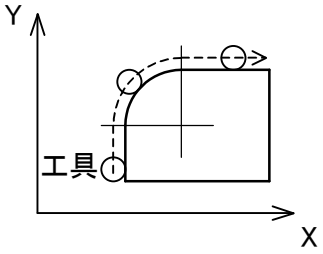
工具を直線または円弧に沿って動かす機能を《**補間機能**》と言います。

①直線に沿った工具の動き



【プログラム指令】
G01Y_F_;
X_Y_;
X_;

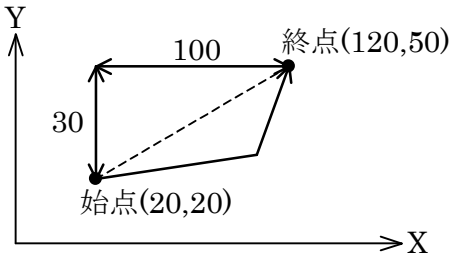
②円弧に沿った工具の動き



【プログラム指令】
G01X_Y_F_;
G02X_Y_R_F_;
G01X_;

４・３・２-1　位置決め(G00)

現在位置から指令された値だけ離れた点へ工具が早送り速度で移動します。
各軸独立に早送り速度で位置決めし、
工具の通路は直線になりません。



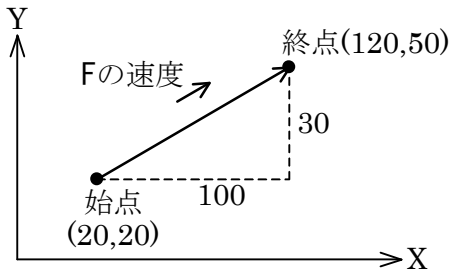
【プログラム例】
G91 G00 X100.0 Y30.0･･･インクリメンタル指令
G90 G00 X120.0 Y50.0･･･アブソリュート指令

> G00 指令の早送り速度設定はパラメータで設定でき、エリアは下記の通りになっています。

パラメータ番号	内容
10	X 軸早送り速度（10 ～ 47000mm/min.）
11	Y 軸早送り速度（10 ～ 47000mm/min.）
12	Z 軸早送り速度（10 ～ 47000mm/min.）
13	A 軸早送り速度（10 ～ 47000mm/min.）

４・３・2-2　直線補間(G01)

現在位置から指定された値だけ離れた点まで、
F で指定された速度で直線に沿って移動します。
F で指定された送り速度は、新たに指定されるまで有効です。
ブロックごとに指定する必要はありません。
F で指定する速度は、直線に沿って動く速度です。
各軸の送り速度は次の通りです。



【プログラム例】
G91 G01 X100 Y30 F200;‥インクリメンタル指令
G90 G01 X120 Y50 F200;‥アブソリュート指令

G01 Xα Yβ Zγ F f;

$$X\text{軸}Fx = \frac{\alpha}{L} \times f$$

$$Y\text{軸}Fy = \frac{\beta}{L} \times f$$

$$Z\text{軸}Fz = \frac{\gamma}{L} \times f \qquad L = \sqrt{\alpha^2 + \beta^2 + \gamma^2}$$

【例】G91 G01 X100.0 Y30.0 F200.0;

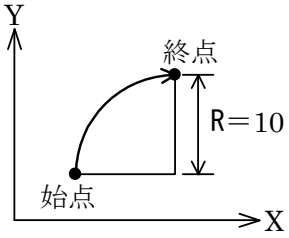
$$\text{分配時間}：\frac{\sqrt{100^2 + 30^2}}{200} \fallingdotseq 0.522\text{min.}$$

$$X\text{軸速度}：\frac{100}{0.522} \fallingdotseq 191.6\text{mm/min.}$$

$$Y\text{軸速度}：\frac{30}{0.522} \fallingdotseq 57.5\text{mm/min.}$$

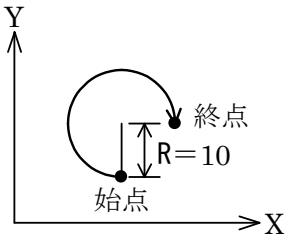
４・３・２-３　円弧補間（G02,03）

現在位置から指令された地点まで、指定された半径値で指定された方向の円弧に沿って工具を動かすことができます。



【プログラム例】（ 180° 以下の円弧）

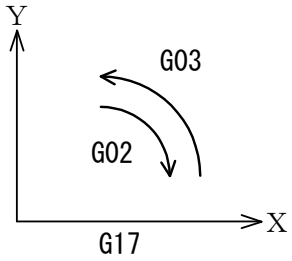
```
G17 G02 X10 Y10 R10 F10;
      ↖方向    ↖半径
```

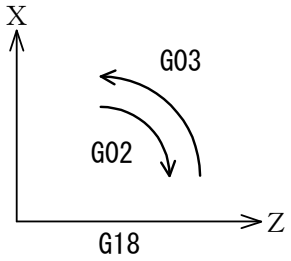


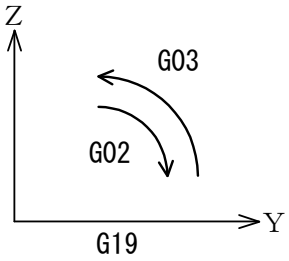
【プログラム例】（ 180° を越え 360° 未満の円弧）

```
G17 G02 X10 Y10 R-10 F10;
      ↖方向    ↖半径（ －指定）
```

円弧補間の送り速度は、F コードに指定された切削速度となり、円弧に沿った速度（円弧の接線方向速度）が指定された送り速度になるように制御されます。





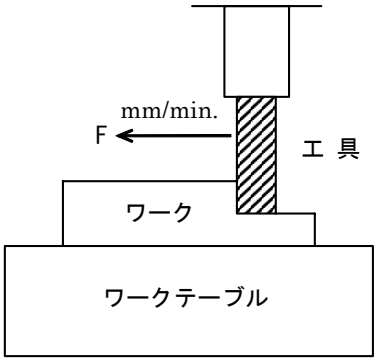


> ⚠ **指令した円弧半径の２倍（直径）が円弧の始点と終点を結ぶ距離より小さい場合、アラーム（円弧半径エラー）が発生します。**
> **360 ° 以上の円弧を１ブロックの命令で動かすことはできません。**

※円弧切削時の半径方向誤差の求め方については、**第 6 章応用編 P. 112 「円弧切削の場合の半径方向誤差について」**を参照してください。

4・3・3　送りについて

ワークを切削するため、指定した速度で工具を動かすことを**《送り》**と言います。
送り速度は、F のあとに続けて実際の数値で指令します。
例えば、工具を 100mm/min. で送りたい時は、プログラムでは F100.0 と指令します。
送り速度を決める機能を**《送り機能》**と言います



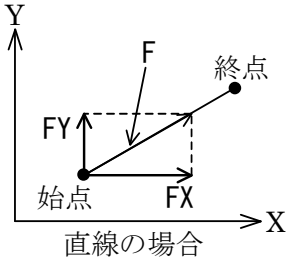
4・3・3-1　早送り

位置決め（G00）の指令により早送りで位置決めされます。
早送り速度は、軸毎にパラメータ［10 ～ 13］が設定されているため、プログラムで指令する必要がありません。

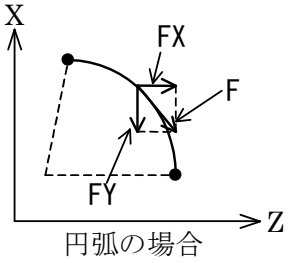
パラメータ番号	内容
10	X 軸早送り速度
11	Y 軸早送り速度
12	Z 軸早送り速度
13	A 軸早送り速度

4・3・3-2　切削送り

直線補間（G01），円弧補間（G02,G03）などで工具をどれくらいの速度で送るのかを F に続く数値で指令します。



直線の場合



円弧の場合

F　：　接線方向の速度

$$F = \sqrt{FX^2 + FY^2}$$

FX　：　X 軸方向の速度成分

FY　：　Y 軸方向の速度成分

４・３・3-3　オーバーライド

①切削送りオーバーライド

CRT 画面上の表示に従い、0 ～ 200%（※外付け押しボタン操作の場合は、0 ～ 150%）のオーバーライドがかけられます。

もし、オーバーライドを上げて行く途中で、合成速度成分が最大速度 47000mm/min. よりも大きくなる場合は、次のような処理をします。

G01 モード ..... 速度過大アラームを表示

G02,G03 ..... 速度過大アラームを表示

②早送りオーバーライド

CRT 画面上の表示に従い、10 ～ 100%（10%ずつ）のオーバーライドかけられます。

③早送りオーバーライドキャンセル

パラメータ［19］を 1 にすると早送りオーバーライドがキャンセルされ、オーバーライドに関係なく早送りは 100%速度になります。

④危険速度警告

G01,G02,G03 モードで、ある軸の速度成分が設定された早送り速度より大きくなる場合は、危険速度の警告が表示されます。

4・3・4　リファレンス点について

位置決めの再現性や繰り返し精度が必要な場合には、各軸に所定の位置を設けます。
この位置を基準にして、機械座標の設定が行われます。この機械上にある固定位置（機械原点）を**《リファレンス点》**と呼びます。
工具をリファレンス点へ移動させる方法には、自動リファレンス点（原点）復帰と手動リファレンス点復帰があります。

4・3・4-1　自動リファレンス点復帰(G28)

G28 X__ Y__ Z__ A__ ･･･ X～A の位置データは中間点の位置指定の指令により、指令した軸を自動的にリファレンス点へ復帰させることができます。
この動作は、まず指令された軸が指令された位置（中間点）まで早送りで位置決めし、指令された全軸が中間点に達した後、更にリファレンス点へ早送りで位置決めを行います。
なお、電源立ち上げ後、一度も手動リファレンス点復帰を行っていない状態では、中間点からの動作は立ち上げ時の位置へ早送りで位置決めを行います。

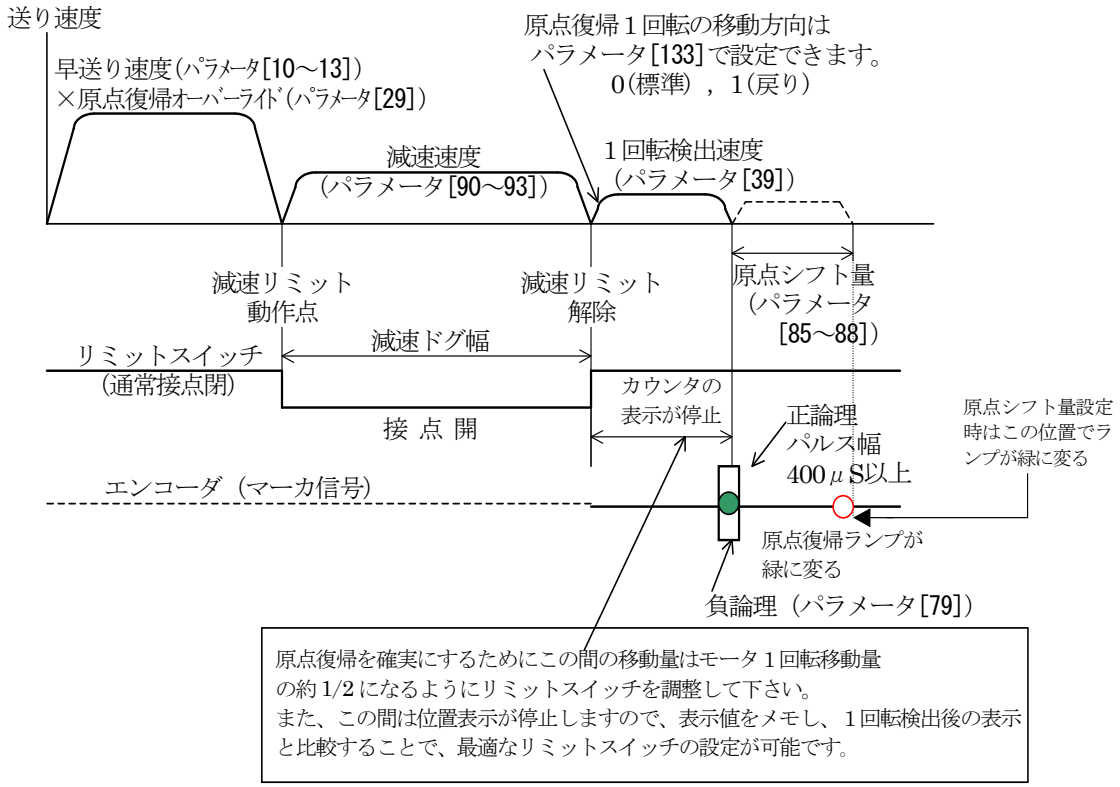
> ⚠ **中間点までの移動は、現在のモードがアブソリュートかインクリメンタルかにより異なりますので御注意ください。**
> **中間点を経由しないで復帰したい場合には、G91 G28 X0 Y0 ･･･ と必ずインクリメンタルモードにしてください。機械原点からスタートする必要があるプログラムの場合は、システム立ち上げ後、最初の 1 回だけは手動かまたは G28.1 ～ G28.4 で各軸の原点復帰操作をしてください。**

G28.1 ～ G28.4 は手動リファレンス点復帰と同じ動作をプログラムで行う指令です。
G28.1 は X、G28.2 は Y、G28.3 は Z、G28.4 は A 軸の原点復帰指令で、同一ブロック内には 1 指令だけプログラムが可能で、他の指令と併用できません。

【例】
G28.3 ..... Z 軸の原点復帰指令（手動リファレンスと同じ方式）
G28.1 ..... X 軸の原点復帰指令（手動リファレンスと同じ方式）
G28.2 ..... Y 軸の原点復帰指令（手動リファレンスと同じ方式）

４・３・４-2　手動リファレンス点復帰

リファレンス点復帰方向に移動させると機械可動部は早送り速度で移動し、リファレンス点復帰減速信号 ＊ DECX ‥‥ ＊ DECA をあげるリミットスイッチを踏むと原点復帰減速速度に減速されます。その後リミットスイッチがはずれますと一定速度に更に減速され、モータ（エンコーダ）からマーカ信号（１回転検出）を検出すると原点シフト量だけ移動して停止し、原点復帰が完了します。



[パラメータ]

10 ..... X 軸原点早送り速度
11..... Y 軸原点早送り速度
12 ..... Z 軸原点早送り速度
13 ..... A 軸原点早送り速度
39 ..... 原点復帰タイミング（１回転検出速度 0 ～ 10）

> **データを小さくすると１回転検出速度は早くなります。**
> **通常は１に設定、原点位置決めを高精度にしたい時や１回転検出信号のパルス幅が短くて検出ミスが発生する場合に大きくします。**

79 ..... モータ１回転マーカ入力信号の論理（負論理＝０，正論理＝１）
85 ～ 88 ..... X ～ A 軸原点シフト量（±10000 パルス）
90 ..... X 軸原点復帰減速速度
91 ..... Y 軸原点復帰減速速度
92 ..... Z 軸原点復帰減速速度
93 ..... A 軸原点復帰減速速度

減速リミットスイッチの設置条件

$$減速ドグ \rightarrow \frac{VR*(100+TS)+VL*(100+TS)}{60000}$$

VR ： 早送り速度　（mm/min.）
TS ： サーボ時定数（msec）
VL ： 原点減速速度（mm/min.）

【例】
VR ＝ 早送り速度（原点復帰速度）＝ 4700mm/min.
TS ＝ サーボ時定数 ＝ 50msec
VL ＝ 原点復帰速度 ＝ 100mm/min.

上記の場合

$$\frac{4700\times(100+50)\ +\ 100\times(100+50)}{60000} = 12mm$$

計算より 12mm 以上減速ドグ幅が必要です。
検出器などの動作時間なども考慮すると、ドグ幅は約 15mm になります。

※減速ドグ幅が不足する場合、パラメータ［29］で原点復帰オーバーライドを設定することにより、原点復帰速度を遅くすることができます。
※各軸のオーバートラベルリミットスイッチの設定も、上記の計算式（ただし VL ＝ 0）でドグ幅を決定してください。
※原点復帰１回転の移動方向を〔戻り〕に設定しますと原点は減速ドグ上になりますので、原点復帰の際に原点復帰と反対方向へ移動させなくて良い利点があります。

4・3・4-3　リファレンス点位置チェック (G27)

G27 のリファレンス点位置チェックは、指令された機械座標上の位置にあるかどうかをチェックする機能です。指令された位置にない場合や、指令された軸が手動リファレンス点復帰をしていない場合、アラームとなります。

【例】G27 X0 Y0 Z0 ･･･ X ～ A の位置データはリファレンス点からの位置

一般に、加工ミスの主な原因はスタート位置が間違っている場合が多く、プログラムのスタート位置にこの指令を入れておきますと、ミス加工の防止に効果があります。

４・３・５　ドウエル(G04)

G04 の指令により指令された時間だけ、次のブロックの動作に移るのを遅らせます。

G04 X10.0　または　G04 P10.0

X ：ドウエルの時間を 0.1sec 単位で設定。(0.1 ～ 99.9sec)

４・３・６　座標系設定

工具の到達すべき位置を、NC システムに教えることによって、NC システムは工具をその位置に移動させます。その到達すべき位置を示すためにある、座標系を設定します。
制御軸が 4 軸の場合、X_Y_Z_A_ と表されます。

座標系には、各々考え方の異なった３通りの設定があります。

①ワーク座標系（絶対座標系）

機械座標の原点から各軸毎にオフセット値をもった座標系で、プログラムの指令はこの座標値に基づいて作成します。

②機械座標系

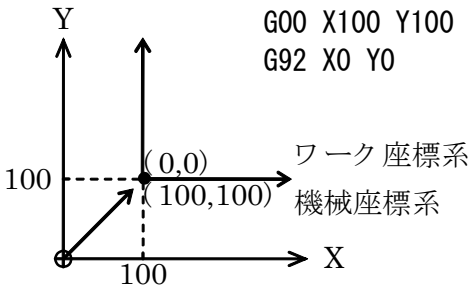
機械固有のリファレンス点を原点とする座標系で、ストロークリミット，原点チェックの場合などに用います。

③相対座標系

この座標系は上記２種類の座標とは別に、全く単独で設定できます。
相対座標を 0 リセット、あるいは任意の数値に設定して、プログラム運転時の移動量確認などに用います。

４・３・６-1　ワーク座標系の設定(G92)

ワーク加工のために使用する座標系を**《ワーク座標系》**または**《絶対座標系》**と呼びます。
この座標系は、次の方法で設立することができます。



G92 を使用する方法

G92 X＿ Y＿ Z＿ A＿

この指令により、工具のある点がこの座標値になるよう、ワーク座標系が確立されます。一度座標系が確立されると、以降のアブソリュート指令はこのワーク座標系での位置になります。

４・３・6-2　機械座標系の設定(G53)

機械座標系とは、ワーク座標とは別に機械固有の座標であり、リファレンス点復帰をすることにより設定されます。
リファレンス点復帰を行わない機械の場合には、電源を立ち上げた位置を（0,0）として座標系が確立されます。
G90 モード（アブソリュート指令）の時にワンショットで有効です。
G91 モード（インクリメンタル指令）の時 G53 指令がありますと、アラームとなります。

４・３・6-3　プログラム指令による相対座標の設定(G193)

相対座標をプログラムにより設定できます。

【例】G193 X1.234 Y2.345 Z0 ･･･ 相対座標値を設定

上記指令は、特定点を基準とした相対座標の表示をさせたい場合や、制御軸拡張で軸切り換えを行った時に相対座標値を変更する場合や、相対座標値が最大表示桁をオーバーする場合に使用します。

４・３・6-4　任意位置原点復帰指令(G128)

原点復帰位置を任意の位置に設定することができます。

【例】G128 X0 Y0 Z0 ････ 現位置を原点にし、X，Y，Z 軸原点復帰完了
　　G128 X1.2 Y3.5 ･･･ 現位置を機械座標 X ＝ 1.2 Y ＝ 3.5 にして、X，Y 軸復帰完了

上記指令は、通常の原点復帰ができない場合や任意の基準点を原点にしたい場合に便利です。

４・３・6-5　相対座標のファイル記録*1(G132,G133)

相対座標をファイル（Data.Txt）に追加記録可能です。

【例】
G133 ･･･ ファイルの内容をクリアします。
G132 ･･･ 相対座標（X，Y，Z，A）の値をファイルに書き込みます。
(P.79 「ティーチング機能」参照)

【記録ファイル例】X123.567Y23.456Z12.345A0.000

この指令とスキップ機能を組み合わせることにより、自動計測システムが容易に構築できます。
Data.Txt の内容をクリアするには G133 を実行してください。

---

*1.NC MATE-4K Ⅱ BASIC では、使用できません。

４・３・７　平面選択（G17,G18,G19）

円弧補間やヘリカル補間を行う平面をＧコードにより選択します。

G17　Ｘ－Ｙ平面（1 軸－２軸）

G18　Ｚ－Ｘ平面（3 軸－１軸）

G19　Ｙ－Ｚ平面（2 軸－３軸）

このコードはモーダルであり、一度指令された平面は次に別の平面が指令されるまで変更されません。なお、G00,G01 の移動指令は、平面選択に無関係です。

４・３・８　アブソリュート指令とインクリメンタル指令（G90,G91）

各軸の移動量を指令する方法として、**《アブソリュート方式》**と**《インクリメンタル方式》**があります。

G90 ：アブソリュート指令

G91 ：インクリメンタル指令

一度 G90 指令があると、G91 指令があるまでアブソリュートモードとなる、モーダルなＧコードです。

①アブソリュート方式（絶対値）

移動の終点位置の座標値をプログラムする方法

「座標系の原点からの距離」即ち指令値は、座標値とみなされます。

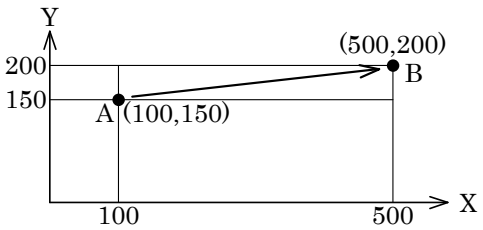
②インクリメンタル方式（相対値）

移動量そのものを直接プログラムする方法

「現在値から次の位置までの距離」即ち指令値は、移動量とみなされます。

両方式の比較

A 点から B 点まで移動する場合の違いを示します。



アブソリュート方式

G90 X500 Y200;

インクリメンタル方式

G91 X400 Y50;

4・4　補助機能（M コード）について

M コードは次の３種類に分けることができます。
①特殊コード
②バイナリ出力コード
③確定出力コード

実際に工具でワークを加工開始するためには、モータを回転させたり、切削油のバルブのオン／オフ制御などが必要で、これら機械側の機器のオン／オフ動作を指令する機能を**《補助機能》**と言い、一般に M コードで指令します。
M コードは、コネクタ CN1 の出力信号を操作する命令で、M7 ～ M0 出力信号（CN1 の 24 ～ 27 番,72 ～ 75 番ピン）の操作を指令します。ただし M コードの M00,M01,M02,M30,M98,M99 は特別な機能が**割当てられています。**

M コードは１ブロックに１つのみ有効です。

【M コードの例】
M03; ..... 主軸正転
M08; ..... 切削油起動
：
M05; ..... 主軸停止
M09 ..... 切削油停止

4・4・1　特殊コード

特殊 M コードは、プログラムの流れを制御するコードです。

コード	名称	意味
M00	プログラムストップ	M00 が指令されますと自動運転を停止します。 スタート信号により自動運転が再開されます。
M01	オプショナルストップ	ブロックスキップ信号が OFF の時、M01 が指令されますと自動運転を停止します。スタート信号により自動運転が再開されます。
M02, M30	エンドオブプログラム	プログラムの終わりを示します。 プログラムの最後には必ず M02 または M30 を指令してください。
M98	サブプログラム呼び出し	サブプログラムを呼び出すのに使用します。
M99	サブプログラムエンド	サブプログラムの終わりを示します。 このコードを実行するとメインプログラムに戻ります。 メインプログラム中で使用するとプログラムの先頭に戻り、プログラム再スタートとして動作します。

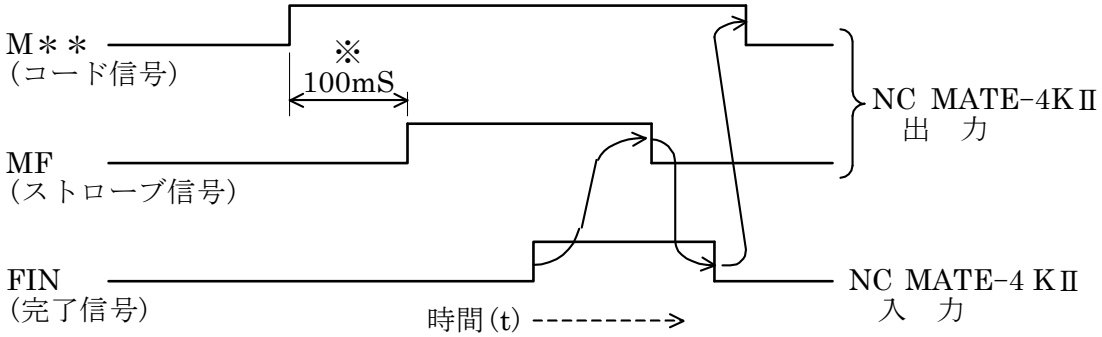
⚠ M コードと移動指令が同一ブロックにプログラムされている場合は、M コードが実行された後に移動指令が実行されます。

４･４･２　バイナリ出力コード

パラメータ 40=0 のとき使用できるコードです。
指令範囲は、M01 ～ M254[*1] です。

このコードは 8 本の M7 ～ M0 出力を一括して設定する命令です。
M に続く数値によって 8 本の M7 ～ M0 出力が決まります。

FIN 信号検出にて動作完了、次のブロックに移ります。
各入出力信号のタイミングチャートは次の通りです。



①コード信号がセットされて、100mS 後にストローブ信号がセットされますので、そのストローブ信号を確認後、コード信号を読み取ってください。
② M コードの機能が終了したら FIN（完了信号）をあげてください。
③ FIN 信号を確認して、ストローブ信号を下げます。
④更に、FIN 信号が停止したのを確認して、コード信号を下げ、次のブロックへ移行します。
(「NC MATE-4K Ⅱ参考回路図」P.30 参照)

※ 100msec. の時間は、パラメータ［130］（**M コード出力タイミング**）によって 0 ～ 1000msec. 単位で設定できます。

*1. 但し特殊コードは、バイナリ出力コードとしては使用できません

４･４･３　確定出力コード

パラメータ 40=1 のとき使用できるコードです。
指令範囲は、M10 ～ M17,M20 ～ M27,M28,M29 です。
このコードは８本の M7 ～ M0 出力のうち１本のみを設定する命令です。

M10 ～ M17 はそれぞれ M0 ～ M7 出力をオンする命令です。
M20 ～ M27 はそれぞれ M0 ～ M7 出力をオフする命令です。
（リセットや読込，無変換，描画，位置復帰の操作で M コード出力は OFF します。）

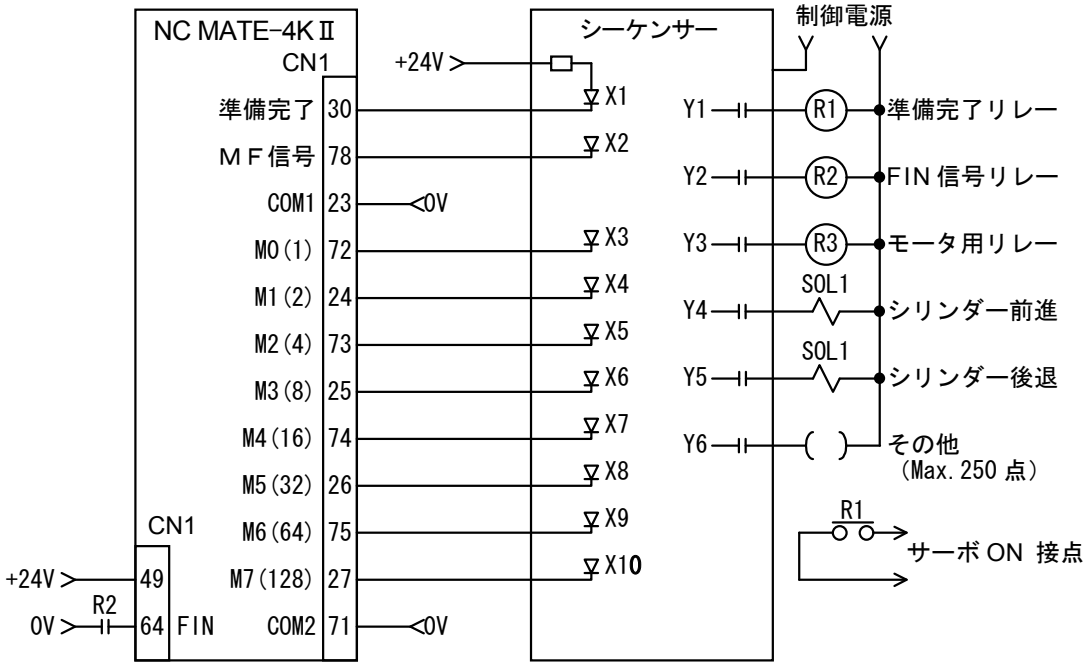
ビット０	M10＝ON ←→ M20＝OFF
ビット１	M11＝ON ←→ M21＝OFF
ビット２	M12＝ON ←→ M22＝OFF
ビット３	M13＝ON ←→ M23＝OFF
ビット４	M14＝ON ←→ M24＝OFF
ビット５	M15＝ON ←→ M25＝OFF
ビット６	M16＝ON ←→ M26＝OFF
ビット７	M17＝ON ←→ M27＝OFF

M28、M29 は特殊な命令です。（動作確認用）
M28 は FIN 信号（CN1 の 64 ピン）がオンするまで待機する命令です。
M29 は FIN 信号（CN1 の 64 ピン）がオフするまで待機する命令です。

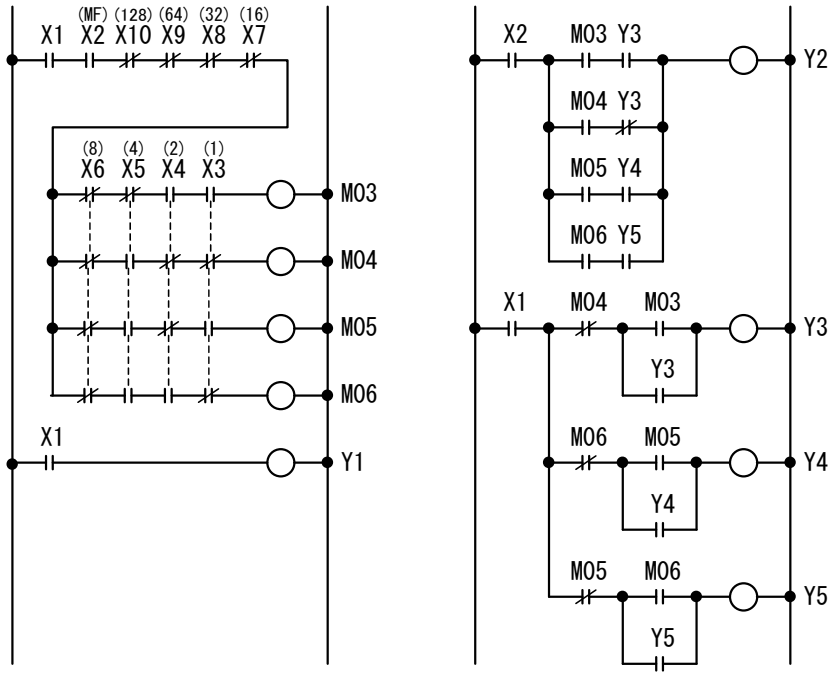
４・４・４　Ｍコード関係参考回路図

４・４・４-1　バイナリＭコードの場合

バイナリ―Ｍコード（約 250 指令，動作確認方式）の場合

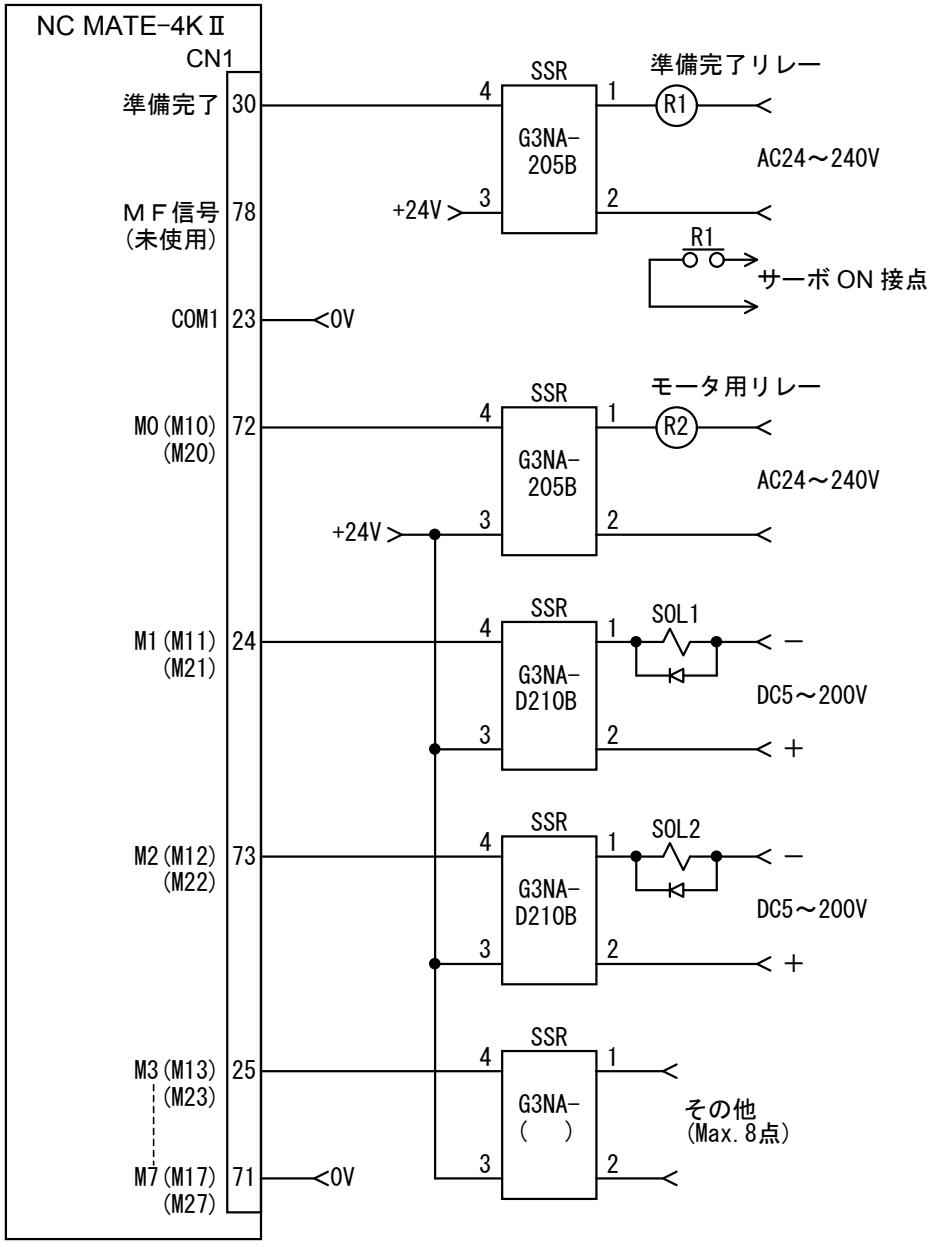


（シーケンサーラダー図）



４・４・４-2　確定Ｍコードの場合

確定Ｍコード（８指令，M10-M17, M20-M27）の場合



注意　外部出力信号は無接点出力で、容量は DC 30V，10mA 以下ですから、リレーやソレノイドは直接駆動できませんので、必ず SSR やシーケンサなどを経由してください。

4･5　工具機能（T コード）について

工具の種類を指令するための機能が、T コードです。
T コードは OUT3 ～ OUT0 出力信号（CN1 の 28,29,76,77 番ピン）の操作をする命令です。
T コードにより４ビット分のデータ（0 ～ 15）を出力することができます。
T コードは M コードとは異なり、ストローブ信号はありませんので、T コードを実行すれば、即ポートの状態（CN1 の 28,29,76,77 番ピン）が変化します。
T コードは１ブロックに１つのみ有効です。

各ビットの状態は下記の通りです。

○：ON　×：OFF

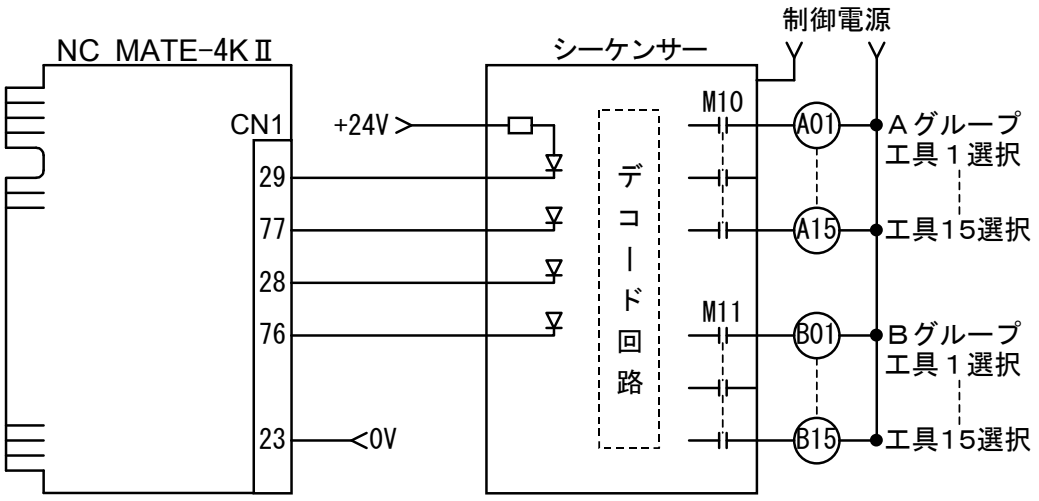
Tコード	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OUT3 (37)	×	×	×	×	×	×	×	×	○	○	○	○	○	○	○	○
OUT2 (38)	×	×	×	×	○	○	○	○	×	×	×	×	○	○	○	○
OUT1 (39)	×	×	○	○	×	×	○	○	×	×	○	○	×	×	○	○
OUT0 (40)	×	○	×	○	×	○	×	○	×	○	×	○	×	○	×	○

T コードを拡張する場合や確認が必要な場合は、M コードと併用することにより最大 16×250 指令まで拡張可能となります。

【プログラム例】

```
T01 ..... T1 コード出力
T02 M10 ..... M コードとの併用
G00 X15 T03 ..... 移動指令と共に出力
G01 Y20 F300 M10 T04 ..... 移動指令と M コードと共に出力
G00 X-15 Y-20 T00 ..... 移動指令と共に出力
M02
```

コード関係参考回路図（Mコードで拡張例）



T コードのポート出力は、次の T コードが実行されるまで前のビット状態を保持します。
但し、M コードと同様、リセット操作等で OFF となります。

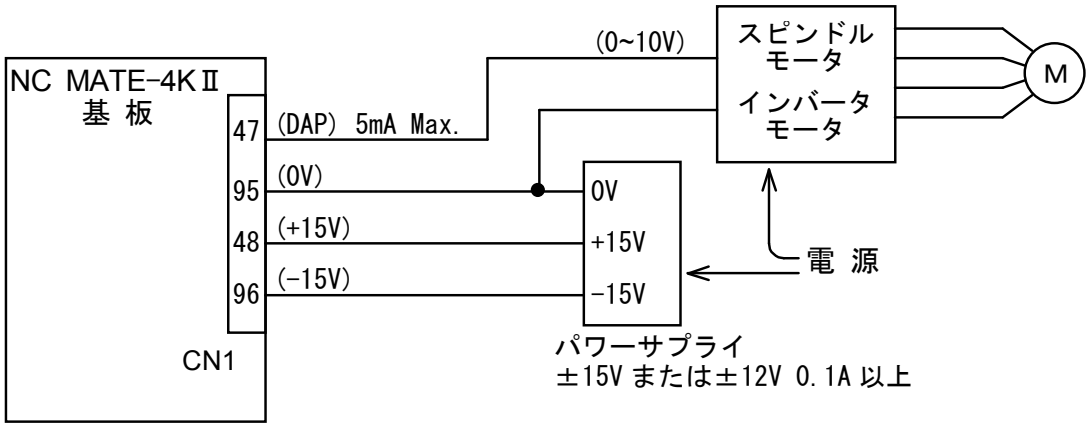
4・6　主軸機能（S コード）について

S コードは DAP アナログ出力信号（CN1 の 47 番ピン）の操作する指令です。
NC MATE-4K Ⅱ基板に +15V、0V、-15V または、+12V、0V、-12V を接続することにより、S コードで 0 ～ 10V のアナログ電圧を出力することができます。
S コードは S0 ～ S100000 の範囲です。
S に続く数値によって DAP 出力に出力する電圧を設定します。

【プログラム例】（パラメータに 4095 を設定した場合）
1）プログラムで S4095 と指令すると 10V が出力されます。
2）プログラムで S1234 と指令すると 1234／4095×10V＝3.013V が出力されます。
3）プログラムで S4100 と 4095 を越える値を指令した場合、S4095 とみなします。
4）プログラムで S-2345 とマイナスの値を指令した場合、S0 とみなします。
5）プログラムで S234.5 と小数点の値を指令した場合、S234 とみなします。

【プログラム例】

;	0V を出力（システム立ち上げ時）
S1234	3.013V を出力
S2345	5.726V を出力
S-2345	0V を出力
S4095	10V を出力
S2048	5V を出力
M02	（次のＳコードが来るまで前の値を保持）



> ⚠ システムが立ち上がる前の出力は、0 ～ 10V の範囲で不定です。

4・7　NC プログラムの作成について

4・7・1　NC プログラム作成要領

メモ帳やワードパットで英数字（大文字）を使用してテキストファイルを作成し
(O1.Txt ～ O9999.Txt) 等のファイル名を付けてプログラムを記録します。
プログラムの第１行目には、タイトル名を記載してください。

■プログラム作成例

(NC MATE-4K Ⅱ CARD CHECK 1) .....（解かり易いタイトル名を１行目に付ける）
G91 .....（G90= 絶対位置指定，G91= 相対位置指定）
M10 .....（スピンドルモータ起動指令。M10 でリレー ON）
G00X100Y100Z100.....（早送りで X，Y，Z 軸正方向移動。）
G01X-100Y-100Z-100F1500 .....（直線補間で負方向移動。合成速度は 1500mm/min）
T03 .....（工具１～ 15 番より 3 番の工具を選択）
G04P1.5 .....（1.5 秒の時間待ち。G04X1.5 でも OK）
G17G02X100Y-100R-100.....（円弧補間で X － Y 平面，時計回り，3/4 円）
G02X-100Y100R100.....（円弧補間で X － Y 平面，時計回り，1/4 円）
G18G02Z100X-100R-100.....（円弧補間で Z － X 平面，時計回り，3/4 円）
G02Z-100X100R100.....（円弧補間で Z － X 平面，時計回り，1/4 円）
G19G02Y100Z-100R-100.....（円弧補間で Y － Z 平面，時計回り，3/4 円）
G02Y-100Z100R100.....（円弧補間で Y － Z 平面，時計回り，1/4 円）
G17G03X-100Y100R100.....（円弧補間で X － Y 平面，反時計回り，1/4 円）
G03X100Y-100R100R-100.....（円弧補間で X － Y 平面，反時計回り，3/4 円）
M20 .....（スピンドルモータ停止指令。M20 でリレー OFF）
/M99.....（スキップ OFF の時、先頭に戻り繰り返し運転）
M02 .....（NC プログラム終了）

上記の例のように、位置決めや直線補間で移動する場合、G00,G01 と移動する軸名と移動距離（単位は mm 少数以下３桁有効）と移動速度（単位は mm/min.）をプログラムします。
尚、(G00,01,F) はモーダル指令ですから、次のステップで変更されない限り有効で、同じコマンドを入力する必要はありません。
位置を指定する方法は、絶対値 =G90 と 相対値 =G91（モーダル指令）があります。
円弧補間で移動する場合、平面指定 G17,G18,G19 と G02 時計回りまたは G03 反時計回りの指定と、移動する軸名 X,Y,Z と移動距離と円弧半径 R をプログラムします。
尚、R の値が負の時は、180 度以上 360 度未満の円弧を指定します。
(G02,G03,G17,G18,G19) はモーダル指令です。
補助機能 M コードのプログラムは、M と数値（10 ～ 17）を指令することにより出力ポートが ON となり、M と数値（20 ～ 27）を指令することで出力ポートが OFF となります。
これで８種類のリレーやソレノイドなどを動作させることにより自動運転が可能となります。

また、パラメータ［40］を変更することにより、250 指令のバイナリ M コードと動作確認（FIN 信号検出）を行う補助機能方式に変更することができます。
時間待ちのプログラムは、G04 と P または X に続く数値（0.1 ～ 99999.9 秒）を指定することにより時間待ちのプログラムが可能となります。

4･7･2　NC プログラム作成上の注意点

1. プログラムの第１行はタイトル名ですから、ここに加工プログラムをした場合は指令が実行されませんので注意してください。

2. プログラムは必ず半角の英数大文字で記載してください。
小文字（x, y, z, a, r, f, g など）や 全角（漢字変換モード）でプログラムを作成しますと、誤動作の原因となりますので注意してください。

【間違い易い文字列】
①０（ゼロ）と Ｏ（オー）
②１（イチ）と 大文字の Ｉ（アイ）と 小文字の ｌ（エル）
③２（ニ）と Ｚ（ゼット）
④ 半角（X,Y,Z,A,Ｆ,G,R･･･）と 全角（X, Y, Z, A, F, G, R ･･･）
⑤ 半角（0, 1, 2, 3 ･･･ 9）と 全角（０，１，２，３･･･９）
⑥ 半角マイナス（-）と 全角（－）
⑦ 半角コンマ（.）と 全角（．）
⑧ 半角スラッシュ（/）と 全角（／）
⑨ 半角（( )）と 全角（（ ））
⑩ 大文字（X, Z, O, P, K ･･･）と 小文字（x, z, o, p, k ･･･）

3. M02 と M30 は、特別な M コードで、NC プログラム終了コマンドになります。

4. M98 と M99 は、サブプログラムの呼出しとリターン指令です。

5. サブプログラムはメインプログラムの後にプログラムしてください。多重ネストは４重まで可能です。

6. 同一ブロック内に同じグループの G コードをプログラムしないでください。
（G0,1,2,3）（G17,18,19）（G43,44,49）（G61,64）（G90,91）は同じグループです。
（G4,27,28,31,43,44,49,53,92,128,132,133,193）の G コードを使用する時は、同一ブロック内に競合する G コードをプログラムしないでください。

7. M コードと移動指令を同一ブロックに指令した場合 M コードの実行後に移動します。

8.NC プログラムに注釈を付ける場合は（ の後にコメントを記載してください。

9. コメントの文字は日本語が使用できますが、移動時の速度変動を避けるため、なるべく半角の英数文字を使用し、短い文章としてください。

10. 文番号は N の後に数字をプログラムできますが、最大４桁までにしてください。
（N1 ～ N9999）

11. システムの立ち上げ後に、一度も手動リファレンス点復帰をしなかった軸を自動原点復帰（G28）した場合、立ち上げ位置を機械原点と見なして原点復帰をいたします。

12. 原点復帰チェック（G27）を使用しますと、原点復帰未完や自動運転開始位置が指定された位置でない場合、アラームとなりますので誤作(ごさく)防止に威力を発揮します。

13. 自動原点復帰（G28）を指令する時は、アブソリュート（G90）とインクリメンタル（G91）モードの違いにより動作が異なりますので、必ず（G90,G91）の指定をプログラムしてください。これらの指令はモーダルなＧコードですから、立ち上げ時はパラメータ［33］で指定された値となっていますが、実行するプログラムによって変化しますので、指定がない場合は以前に実行したモードとなり、誤動作の原因となります。

14. 平面指定（G17,G18,G19）も立ち上げ時はパラメータ［32］で指定された値となっていますが、平面指定が前のプログラムで変更された場合、プログラム中に指定がない時は平面指定が以前に実行したモードとなり、誤動作の原因となります。

15. 位置指定（G0,1,2,3,90,91）や 平面指定（G17,18,19）などモーダル指令は、立ち上げ時にパラメータで指定された値となっていますが、前のプログラムで変更された時指定が無いと加工ミスの原因となりますので、必ずプログラムしてください。

16. 工具径補正[*1]（G41,G42）を指令した場合には工具径補正キャンセル（G40）、工具長補正（G43, G44）を指令した場合には工具長補正キャンセル（G49）を必ずペアーで指令するように注意してください。

17. スケーリングやミラーイメージ，座標回転も同様にキャンセルをプログラムしてください。

*1. NC MATE-4K Ⅱ BASIC では、使用できません。

第５章　機能編

５・１　パソコン NC 開発キットの各種機能について*1

パソコン NC 開発キットの NC 制御ソフトは、NC の基本的な機能を内蔵しているので、G コードを使用した数値制御システムが容易に構築できます。

パラメーター一覧表

機能	内容
手動ハンドル制御機能	手動パルスハンドルを使って出力パルスの微調整が行えます。
手動介入機能	自動運転中にプログラムによる移動に加えて、手動ハンドルによる移動を介入させる機能です。
工具補正機能	工具径補正および工具長補正が行えます。
スキップ機能	指令された移動ブロックを、スキップ信号により中断させ次のブロックへ進ませることができます。
遠隔制御運転機能	ホスト側で作成したデータをインターネットや LAN などで転送することにより、遠隔制御による自動運転が実行できます。
ティーチング機能	手動操作で軸移動を行い、その相対座標をファイルに追加記録できます。
ヘリカル補間機能	円弧補間と同期して動く直線軸を指定することにより、螺旋状に移動することができます。
ピッチ誤差補正機能	各軸ごとにピッチ誤差の補正ができます。
ユーザーマクロリンク機能	C 言語や BASIC などを使用して開発されたユーザーのプログラムとリンクすることができ、変数や演算機能を必要とする高度な制御が可能です。
イグザクトストップ機能	プログラムのブロック間にパラメータで設定された時間を設ける機能です。
外付け押ボタン操作機能	外付け押ボタン操作方式に変更することが可能です。
スケーリング機能	加工プログラムで指令した形状の縮小、拡大ができます。
プログラマブルミラーイメージ機能	加工プログラムで指令した形状を反転させることができます。
座標回転機能	加工プログラムで指令した形状を回転させることができます。
円弧補間 IJK 指定機能	円弧補間を半径 R ではなく、中心までの距離（I,J,K の内の 2 軸）を使って表示することができます。
ファイル連結機能	大容量ファイルを運転させることができます。
クッション機能	サーボモータやパルスモータの加減速のショックを少なくすることができます。

*1.　　は、NC MATE-4K Ⅱ BASIC では、使用できません。

パラメータを設定することにより、各種機能を有効にすることができます。

番号	内容	設定		設定範囲	出荷時設定
[0]	X 軸バックラッシュ量（パルス）			0 ～ 255	0
[1]	Y 軸バックラッシュ量（パルス）			0 ～ 255	0
[2]	Z 軸バックラッシュ量（パルス）			0 ～ 255	0
[3]	A 軸バックラッシュ量（パルス）			0 ～ 255	0
[4]	現在未使用				
[5]	X 軸原点復帰方向	0	＋方向		0
		1	－方向		
[6]	Y 軸原点復帰方向	0	＋方向		0
		1	－方向		
[7]	Z 軸原点復帰方向	0	＋方向		0
		1	－方向		
[8]	A 軸原点復帰方向	0	＋方向		0
		1	－方向		
[9]	ピッチ誤差補正	0	無効		99
		99	有効		
[10]	X 軸早送り速度			10 ～ 47000mm/min.	10000
[11]	Y 軸早送り速度			10 ～ 47000mm/min.	10000
[12]	Z 軸早送り速度			10 ～ 47000mm/min.	10000
[13]	A 軸早送り速度			10 ～ 47000mm/min.	10000
[14]	現在未使用				
[15]	X 軸回転方向	0	正方向		0
		1	逆方向		
[16]	Y 軸回転方向	0	正方向		0
		1	逆方向		
[17]	Z 軸回転方向	0	正方向		0
		1	逆方向		
[18]	A 軸回転方向	0	正方向		0
		1	逆方向		
[19]	早送りオーバーライド	0	有効		0
		1	無効		
[20]	現在未使用				
[21]	現在未使用				
[22]	現在未使用				
[23]	現在未使用				
[24]	IJK －＞ R 変換	0	無効		3
		3	有効		
[25]	終了確認ダイアログ	0	表示		1
		1	非表示		

番号	内容	設定		設定範囲	出荷時設定
[26]	表示軸	+1	X 軸	1 ～ 15	15
		+2	Y 軸		
		+4	Z 軸		
		+8	A 軸		
[27]	制御ソフト画面レイアウト *1)	0			1
		1			
[28]	MDI 入力データクリア	0	クリアなし		0
		1	クリアあり		
[29]	原点復帰オーバーライド%			10 ～ 100	50
[30]	グループ 01 の G コード初期値	0	G00		1
		1	G01		
[31]	スキップ信号の論理	0	機能なし		31
		31	正論理		
		32	負論理		
[32]	グループ 03 の G コード初期値			17、18、19	17
[33]	グループ 04 の G コード初期値			90、91	91
[34]	X 軸直径指定	0	半径		0
		1	直径		
[35]	スケーリング変換	0	無効		51
		51	有効		
[36]	S コードアナログ出力設定値			10 ～ 100000	1000
[37]	ミラーイメージ変換	0	無効		511
		511	有効		
[38]	座標回転変数	0	無効		68
		68	有効		
[39]	原点復帰タイミング（1 回転検出速度）			0 ～ 10	1
[40]	M コード体系	0	バイナリ		1
		1	確定コード		
[41]	設定単位	0	0.001mm（固定）		0
[42]	現在未使用				
[43]	背景色			0、1、2、3、4、5、6、7、8、9、10	0
[44]	現在未使用				
[45]	現在未使用				
[46]	現在未使用				
[47]	軌跡シミュレーション描画	0	無効		17
		17	有効		
[48]	ティーチング機能	0	無効		99
		99	有効		
[49]	手動介入機能	0	無効		104
		104	有効		
[50]	現在未使用				

番号	内容	設定		設定範囲	出荷時設定
[51]	イグザクトストップ			待ち時間方式 0 ～ 9999ms 位置検出方式 10000	100
[52]	干渉チェック	0	有り		0
		1	無し		
[53]	工具補正機能（工具長＆工具径）	0	無効		20
		20	有効		
[54]	円弧半径 R の誤差範囲（パルス）			0 ～ 100	1
[55]	スキャン	0	40.96		1
		1	20.48		
[56]	外付押ボタン操作機能	0	無効		0
		30	正論理		
		31	負論理オーバーライド		
[57]	ヘリカル補間機能	0	無効		8
		8	有効		
[58]	現在未使用				
[59]	基板仕様		2		2
[60]	X 軸荒 JOG 速度			10 ～ 2000mm/min	500
[61]	Y 軸荒 JOG 速度			10 ～ 2000mm/min	500
[62]	Z 軸荒 JOG 速度			10 ～ 2000mm/min	500
[63]	A 軸荒 JOG 速度			10 ～ 2000mm/min	500
[64]	現在未使用				
[65]	X 軸ソフトリミット－方向	0	無効	1 ～ 9999999mm	0
[66]	Y 軸ソフトリミット－方向	0	無効	1 ～ 9999999mm	0
[67]	Z 軸ソフトリミット－方向	0	無効	1 ～ 9999999mm	0
[68]	A 軸ソフトリミット－方向	0	無効	1 ～ 9999999mm	0
[69]	原点復帰後の ＯＴ リミット SW 無視	0	無効		0
		1	有効		
[70]	X 軸細 JOG 速度			10 ～ 1000mm/min	50
[71]	Y 軸細 JOG 速度			10 ～ 1000mm/min	50
[72]	Z 軸細 JOG 速度			10 ～ 1000mm/min	50
[73]	A 軸細 JOG 速度			10 ～ 1000mm/min	50
[74]	早送り描画削除	0	無効		0
		20	有効		
[75]	X 軸ソフトリミット＋方向	0	無効	1 ～ 9999999mm	0
[76]	Y 軸ソフトリミット＋方向	0	無効	1 ～ 9999999mm	0
[77]	Z 軸ソフトリミット＋方向	0	無効	1 ～ 9999999mm	0
[78]	A 軸ソフトリミット＋方向	0	無効	1 ～ 9999999mm	0

番号	内容	設定		設定範囲	出荷時設定
[79]	エンコーダ１回転の論理	+1	X 軸	(X,Y,A= 負論理 Z= 正論理の例は 4)	0
		+2	Y 軸		
		+4	Z 軸		
		+8	A 軸		
[80]	X 軸微 JOG 速度			5 ～ 50mm/min.	5
[81]	Y 軸微 JOG 速度			5 ～ 50mm/min.	5
[82]	Z 軸微 JOG 速度			5 ～ 50mm/min.	5
[83]	A 軸微 JOG 速度			5 ～ 50mm/min.	5
[84]	非常停止で原点復帰完了 ＯＦＦ	0	無効		1
		1	有効		
[85]	X 軸原点シフト量（パルス）			－ 10000 ～ 10000	0
[86]	Y 軸原点シフト量（パルス）			－ 10000 ～ 10000	0
[87]	Z 軸原点シフト量（パルス）			－ 10000 ～ 10000	0
[88]	A 軸原点シフト量（パルス）			－ 10000 ～ 10000	0
[89]	（オプションパラメータ）0				0
[90]	X 軸原点復帰減速速度			10 ～ 1000mm/min	200
[91]	Y 軸原点復帰減速速度			10 ～ 1000mm/min	200
[92]	Z 軸原点復帰減速速度			10 ～ 1000mm/min	200
[93]	A 軸原点復帰減速速度			10 ～ 1000mm/min	200
[94]	早送り加減速時間			10 ～ 1000ms	60
[95]	X 軸パルス出力方式	0	正パルス逆パルス方式		0
		1	パルス列方向方式		
		2	A 相 B 相方式		
[96]	Y 軸パルス出力方式	0	正パルス逆パルス方式		0
		1	パルス列方向方式		
		2	A 相 B 相方式		
[97]	Z 軸パルス出力方式	0	正パルス逆パルス方式		0
		1	パルス列方向方式		
		2	A 相 B 相方式		
[98]	A 軸パルス出力方式	0	正パルス逆パルス方式		0
		1	パルス列方向方式		
		2	A 相 B 相方式		
[99]	現在未使用）				

番号	内容	設定		設定範囲	出荷時設定
[100]	現在未使用)				
[101]	現在未使用)				
[102]	マウス正逆寸時に外付正逆寸も有効	0	無効		0
		1	有効		
[103]	リセット時の M.T 出力クリア	0	クリア		0
		1	保存		
[104]	モーダル G104 設定 G100 でクリア	0	無効		0
		1	有効		
[105]	位置復帰の優先軸	0	Z		0
		1	X		
		2	Y		
		3	A		
		4	なし		
[106]	ピッチ誤差補正間隔	0	10mm		0
		1	1mm		
[107]	MDI 令初期値	0	なし	0 ～ 1000	0
		1	G200R1		
		2	G200R2		
[108]	現在未使用				
[109]	現在未使用				
[110]	描画平面初期値	0	X-Y 平面		0
		1	Z-X 平面		
		2	Y-Z 平面		
[111]	現在未使用				
[112]	現在未使用				
[113]	現在未使用				
[114]	現在未使用				
[115]	X 軸クッション機能加減速時定数			0 ～ 1000ms	0
[116]	Y 軸クッション機能加減速時定数			0 ～ 1000ms	0
[117]	Z 軸クッション機能加減速時定数			0 ～ 1000ms	0
[118]	A 軸クッション機能加減速時定数			0 ～ 1000ms	0
[119]	現在未使用				
[120]	X 軸電子ギア分子			1 ～ 65535	1
[121]	Y 軸電子ギア分子			1 ～ 65535	1
[122]	Z 軸電子ギア分子			1 ～ 65535	1
[123]	A 軸電子ギア分子			1 ～ 65535	1
[124]	現在未使用				
[125]	X 軸電子ギア分母			1 ～ 65535	1
[126]	Y 軸電子ギア分母			1 ～ 65535	1
[127]	Z 軸電子ギア分母			1 ～ 65535	1

番号	内容	設定		設定範囲	出荷時設定
[128]	A 軸電子ギア分母			1 ～ 65535	1
[129]	現在未使用				
[130]	M コード出力タイミング			0 ～ 1000（ms）	100
[131]	ファイル連結機能	0	無効		200
		200	有効		
[132]	現在未使用				
[133]	原点復帰 1 回転移動方向	0	標準		0
		1	戻り		
[134]	工具補正選択の初期値	0	R		0
		1	X		
		2	Y		
		3	Z		
		4	A		
[135]	表示座標初期値	0	相対		0
		1	絶対		
		2	機械		
		3	残量		
[136]	モード選択初期値	0	原点		1
		1	早		
		2	粗		
		3	細		
		4	微		
		5	ハンドル		
[137]	軸選択初期値	0	X 軸		0
		1	Y 軸		
		2	Z 軸		
		3	A 軸		
[138]	ハンドル倍率選択初期値	0	1 倍		0
		1	10 倍		
[139]	ブロックスキップ選択初期値	0	非選択		1
		1	選択		
[140]	ユーザーマクロリンク機能	0	無効		400
		400	有効		
[141]	現在未使用				
[142]	ファイル記録座標	0	相対		0
		1	絶対		
		2	機械座標		
[143]	記録軸選択	+1	X 軸		15
		+2	Y 軸		
		+4	Z 軸		
		+8	A 軸		
		+16	B 軸		

番号	内容	設定		設定範囲	出荷時設定
[144]	記録ドライブ	0	カレント		0
		232	C		
		233	D		
		234	E		
		235 ～ 239	F ～ J		
[145]	遠隔制御運転	0	無効		0
		232	C		
		233	D		
		234	E		
		235 ～ 239	F ～ J		
[146]	遠隔制御初期値	0	OFF		0
		1	チェック マーク ON		
[147]	マウスクランプ解除	0	無効		0
		1	有効		
[148]	手動ハンドル 10 倍	0	無効		0
		1	有効		
[149]	基板クロック	0	0.8M		1
		1	1.6M(固定)		
		2	3.2M		
		3	6.4M		
[150]	現在未使用				
以下現在未使用					

①パラメータを変更された場合は、制御ソフトを一度終了させ、再度立ち上げて御使用ください。
②ハードディスクの寿命（２～５年）に対応するため、必ず最終パラメータ値を CD-ROM などに保存し、取扱説明書に記録して保管してください。
このシステムを最終ユーザに納品される場合は、必ず制御ソフト（最終パラメータ記録）をコピーし、最終ユーザ資料に添付してください。

5・2　NC 制御ソフト仕様一覧

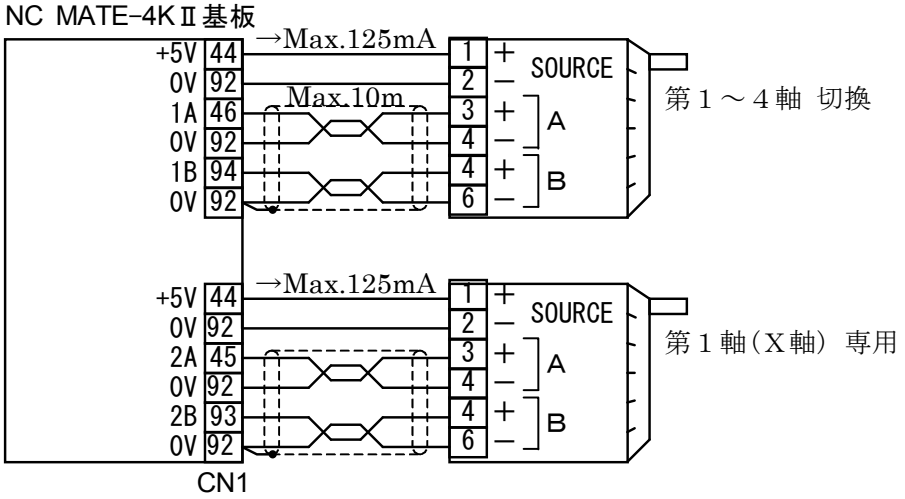
仕様	内容
同時制御軸数	4 軸
補間指令	直線補間、多象限円弧補間
NC 指令	ワードアドレス　フォーマット
最大ストローク	± 9999999.999mm
最大指令値	9999999.999mm
最大送り速度	47m/ 分
オーバーライド	10 → 200%（＊外付け押しボタン操作の場合は、10 → 150%）
入力ポート	40 ポート
出力ポート	16 ポート
登録プログラム	9999 本
最小設定単位	0.001mm
ステップ数	399999 ステップ
最大文字数	60 文字 / ステップ
使用可能 G コード*i)	G00,01,02,03,04,09,10,17,18,19,27,28,28.1,28.2,28.3,28.4,31,31.1,40,41,42,43,44,49,50,50.1,51,51.1,53,61,64,68,69,90,91,92,104,128,132,133,193,200,300,400
サブプログラム	多重ネスティング可能（実行前にサブプログラム展開する方式）
補助機能（M）	8 指令または 250 指令（動作確認を行う補助機能方式）
工具機能（T）	15 指令、M コードと併用で拡張が可能（15 ＊ 250 指令）
主軸機能（S）	± 15V または± 12V 電源入力で S コードアナログ出力が可能
手動ハンドル	手動パルスハンドル追加で、同時 2 軸のハンドル送りが可能
バックラッシュ	補正量（0 ～ 255 パルス）
電子ギア	ギア比（1/10 ～ 10）、データ（1 ～ 65535）
実行速度	約 100 ステップ / 秒

＊i)NC MATE-4K Ⅱ BASIC では、一部使用できません。

５・３　手動ハンドル制御機能

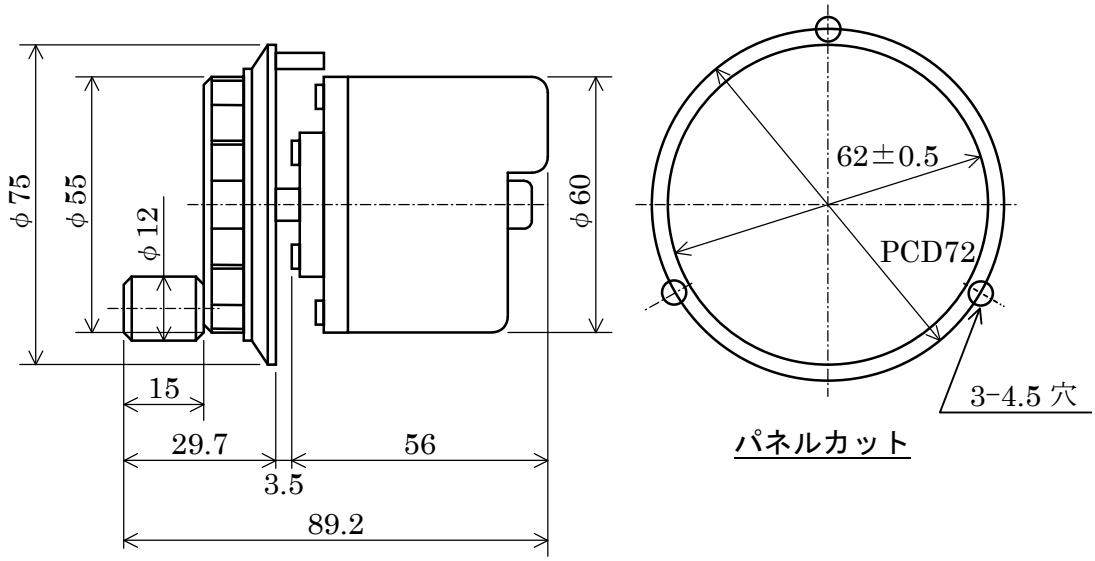
手動パルスハンドルを使って、出力パルスの微調整を行うことができます。

５・３・１　サムタク製手動パルスハンドルの接続方法



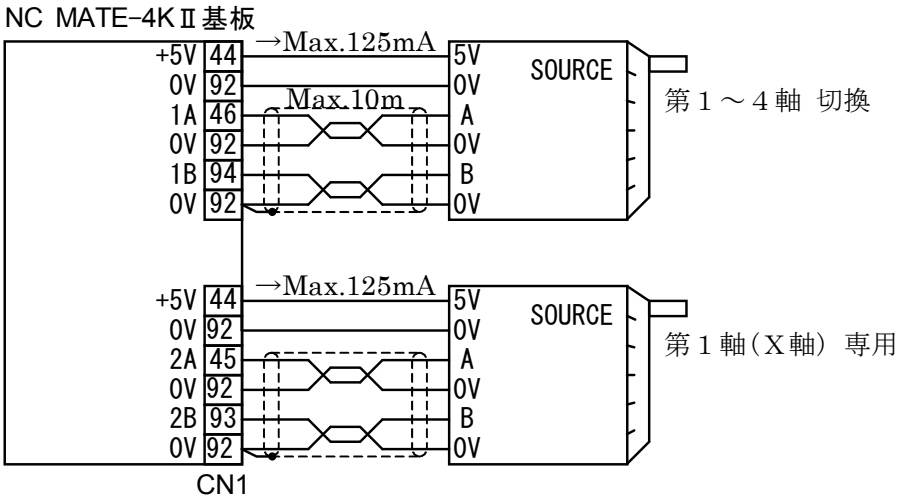
※ 同時２軸の軸移動を必要としない場合、第１軸専用の接続は不要です。

５・３・２　サムタク製手動パルスハンドル外形図



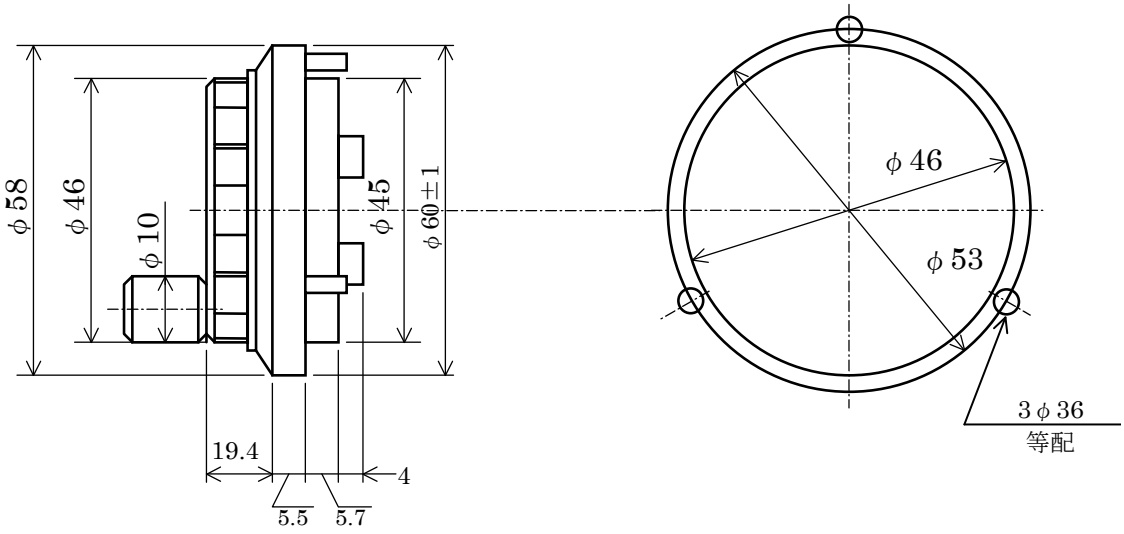
手動パルスハンドル サムタク　HGF-031-100 相当（消費電流 50mA） ※接続参考製品です。ご購入前に型番等最新情報をメーカーにお問い合わせください。

５・３・３　ネミコン製手動パルスハンドルの接続方法



※同時２軸の軸移動を必要としない場合、第１軸専用の接続は不要です。

５・３・４　ネミコン製手動パルスハンドル外形図



パネルカット

手動パルスハンドル ネミコン　UFO-M2-01-2Z9-B00E 相当（消費電流 70mA）
※当社オプション品：ε PH-UFO-M2-01-2Z9-B00E ※取り付けの際には、最新情報をメーカーホームページでご確認ください。

５・４　手動介入機能*1

この機能は、自動運転中にプログラムによる移動に加えて、手動ハンドルによる移動を介入させる機能で、相対，絶対，機械座標が変化します。
手動介入は、早送りと直線補間、円弧補間（G00,01,02,03）実行中でのみ可能で、介入させるブロックに G104（**手動介入**）を入れることにより、ワンショットで有効となります。

Gコード	内容
G104	手動介入

【手動介入のプログラム例】

```
G91G00X-20Y50          （早送りで位置決め介入無効）
G104G01X50Z-1F90       （直線補間介入有効）　……①
G104G17G02X10Y10R10    （円弧補間介入有効）……②
G104G00X-40Y-60Z1      （早送り介入有効）　……③
```

上記プログラムで、ブロック①の自動運転中に X 軸を 0.1mm 手動介入した時の座標は、下記のようになります。
ブロック① 終了時の座標　X＝30.1　Y＝50　Z＝-1
ブロック② 終了時の座標　X＝40.1　Y＝60　Z＝-1
ブロック③ 終了時の座標　X＝0.1　Y＝0　Z＝0

【アブソリュートでプログラムした例】

```
G90G00X-20Y50          （早送りで位置決め介入無効）
G104G01X30Z-1F90       （直線補間介入有効）　……①
G104G17G02X40Y60R10    （円弧補間介入有効）……②
G104G00X0Y0Z0          （早送り介入有効）……③
```

上記と同じ手動介入した座標は、下記のようになります。
ブロック① 終了時の座標　X＝30.1　Y＝50　Z＝-1
ブロック② 終了時の座標　X＝40　Y＝60　Z＝-1
ブロック③ 終了時の座標　X＝0　Y＝0　Z＝0

※ブロック①の実行中に、フィードホールドで自動運転を休止させて、手動で移動させた場合も、上記と同じ動作をします。G104 指令は、自動運転中に手動介入を有効にする指令ですから、休止中の場合は、G104 指令がなくても上記動作をします。
※手動介入後のブロックで、座標を介入量だけシフトさせた場合は、G91 モードでプログラムを作成して下さい。
※自動運転中に軸選択やハンドル倍率を外部スイッチ等で変更しても、自動運転を休止しないと変更されませんので、注意してください。
※パラメータ［104］を１にすると、G104 指令はワンショットからモーダルに変更できます。
G104 で介入開始、G100 でキャンセルですが、介入が連続する特別な用途でのみ使用してください。

---

*1. NC MATE-4K Ⅱ BASIC では、使用できません。

５・５　工具補正機能*1

パラメータの変更により、準備機能（G40,G41,G42）を用いて工具径補正、同じく（G43,G44,G49）を用いて工具長補正ができます。

５・５・１　工具補正の概要

◆工具径補正機能とは、プログラムされた通路よりカッター径だけオフセットした通路を、自動的に作成する機能のことです。G41 指令で工具進行方向の左側オフセット，G42 指令で右側オフセット，G40 指令でキャンセルします。
補正データは 32 組（D01 ～ D32）で、半径値（0 ～ ±9999.999mm）で指定ができます。

◆工具長補正機能とは、プログラムされた通路より補正データに設定された値だけ、＋あるいは－側にずらす機能のことです。H コードで指定された番号の X，Y，Z，A 軸データを G43 指令で＋側，G44 指令で－側にオフセット，G49 指令でキャンセルします。
補正データは、32 組（H01 ～ H32）で（0 ～ ±9999.999mm）が指定できます。

◆工具径補正方式は、実行中に計算して補正する方式でなく、プログラム読み込み編集時に補正計算を行う　方式を採用していますので、ブロック間の実行速度が速く、干渉チェックが実行前にできる特長があります。
更に、工具径補正後のプログラムを見ることができますので、工具通路の詳細なチェックが可能です。ただし、工具径を変更した場合は、再度読み込み、補正計算を行う必要があります。

◆工具補正の干渉チェック解除は、パラメータで選択できます。干渉アラームが発生する場合、干渉チェック解除にして補正計算を行い、補正計算後のデータを別ファイルに登録し、適宜修正編集をすることにより、通常であれば干渉チェックにより、工具補正が不可能なプログラムでも使用することができます。

◆工具長補正方式は、実行中に補正する方式を採用しており、その補正量は工具長位置オフセットを変化させますので、絶対座標は変化しません。
頻繁に工具長を補正しても、すぐにプログラムを実行させて実際の移動量を確認することができます。

> ⚠ **干渉チェックとは、径補正の結果、素材に切り込み過ぎが発生することを未然に防ごうとする機能ですが、全ての干渉が防げるわけでなく、また実際には干渉しないけれどもチェックされる場合もあります。**

*1.NC MATE-4K Ⅱ BASIC では、使用できません。

５・５・２　工具補正コード一覧表

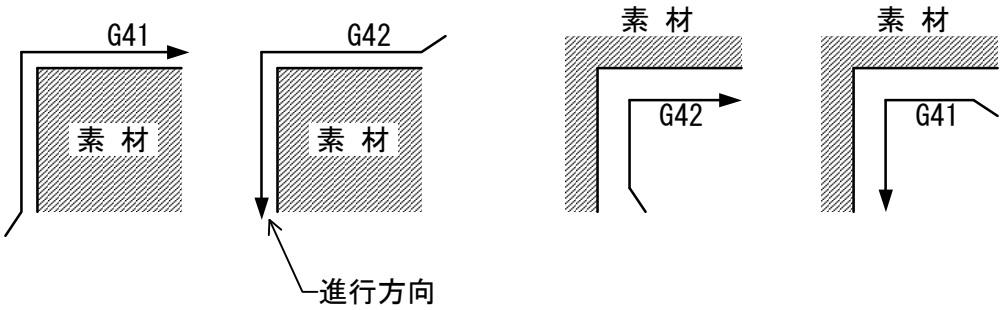
表 5-1: 平面指定

Ｇコード	径補正平面
G17	XY 平面
G18	ZX 平面
G19	YZ 平面

Ｇコード	径補正平面
G40	工具径補正キャンセル
G41	工具進行方向の左側オフセット
G42	工具進行方向の右側オフセット
G43	工具長＋側オフセット
G44	工具長－側オフセット
G49	工具長キャンセル

No.	径補正（Ｄ）	長補正（Ｈ）
01	0 ～ 9999.999	0 ± 9999.999
02	・ ・	・ ・
・ ・	・ ・	・ ・
・ ・	・ ・	・ ・
32	・ ・	・ ・

【工具径補正方向例】



５･５･３　工具径および工具長補正上の注意事項

◆工具径および長補正のスタートおよび終了は、位置決め（G00）あるいは直線補間（G01）モードでなければなりません。

◆ G40,G41,G42 指令は、移動指令とともにプログラムされていなければなりません。
（G43,G44 指令は、移動指令なしでプログラムできます。）

◆補正に必要な指令（G90,G91,G17,G18,G19,G00,G01）や補正データ番地（D01 ～ D32, H01 ～ H32）が指定さ　れていなければなりません。

◆径補正中に平面指定（G17 ～ G19）の変更や、リファレンス点チェック（G27），自動原点復帰（G28,G128），スキップ（G31,G31.1），座標系設定（G53,G92,G193）をプログラムしてはいけません。

◆径補正データ番地の変更（D01 ～ D32）や、径補正方向の変更（G41、G42）は、補正キャンセル（G40）を経由して下さい。

◆径補正データを変更した場合は、再度プログラムを入力し、補正計算を行って下さい。
G10 指令で工具径を変更した場合も、再度プログラムを読み込み、補正計算を行って下さい。

◆径補正のプログラム中移動を伴わないブロックを、５ブロック以上連続して指令してはいけません。また、径補正平面での軸移動指令が必要です。

◆ MDI からの入力指令には、工具径補正は一切行われません。

◆干渉チェックにてアラームが出た場合や、径補正後のプログラムが適当でない場合は、工具データを変更するかまたは、プログラム通路を変更して下さい。

◆正確に径補正の変換ができない場合は、補正計算後のデータを別ファイルに登録し、適宜修正編集をして使用してください。

◆工具長補正（G43,G44）をプログラムした場合は、工具長補正キャンセル（G49）を忘れずにプログラムして下さい。

◆工具長補正のデータ（H01 ～ H32）の変更は、工具長補正キャンセルモードで行って下さい。

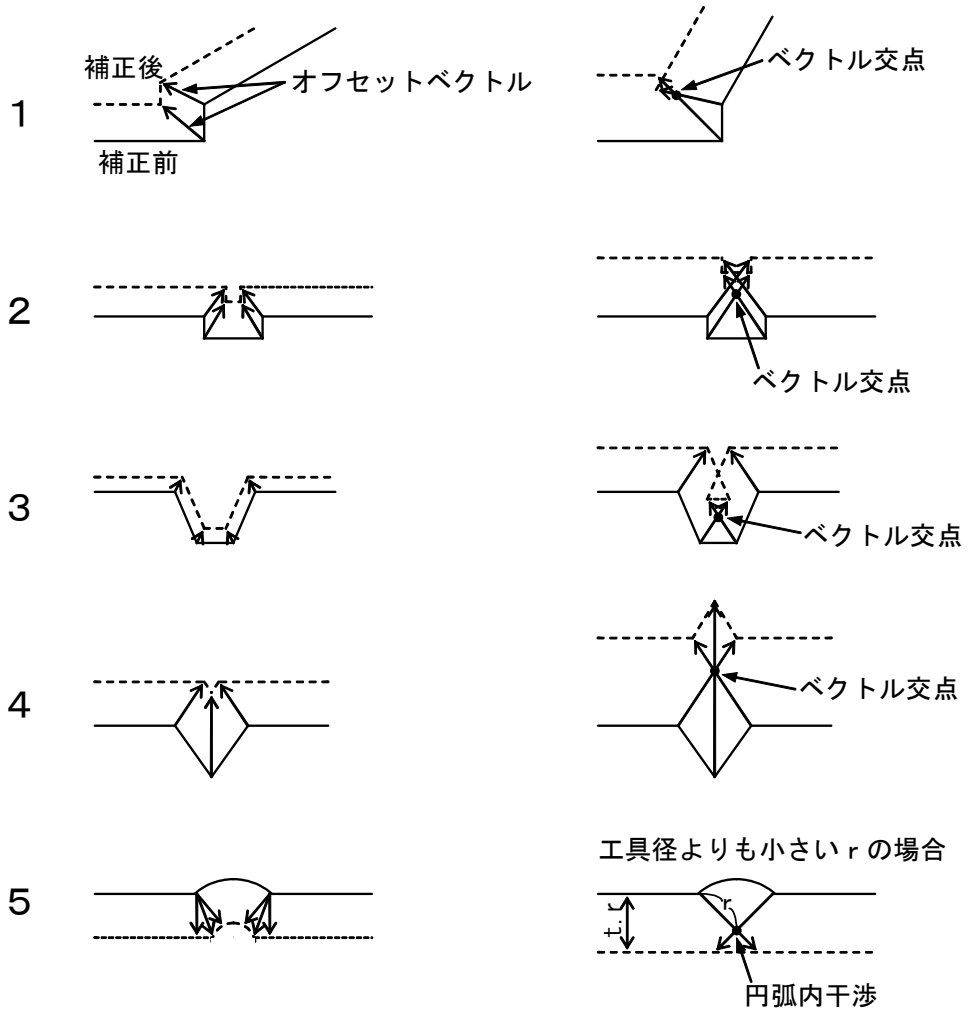
◆工具長補正ベクトルは、工具長補正キャンセル（G49），原位置復帰，プログラムの読み込みリセット操作，軌跡描画の実行によりキャンセルされます。

◆正確に変換できない場合は、手入力にて修正してください。

５・５・４　工具径補正干渉チェック

干渉とは、工具が材料を切り込んでしまう場合を言い、そのチェックは、１ブロック毎のオフセットベクトルが交差するかどうかで判断します。

〔干渉しない例〕　　〔干渉する例〕



５・５・５　プログラムによる工具補正量の変更

工具径および工具長オフセット量をプログラムにより入力することができます。

G90 G10 P_ R_ ……（工具径）（必ず G90 または G91 を指定してください。）

G91 G10 P_ X_ Y_ Z_ A_ ……（工具長）

G90 G10 P_ R_ X_ Y_ Z_ A_ ……（工具径および工具長）

P：工具オフセット番号　　R：工具径オフセット量　　X ～ A：工具長オフセット量

工具径オフセット量 R および工具長オフセット量 X ～ A はアブソリュートでもインクリメンタルでも指令できます。G90 モードで G10 を指令した場合、オフセット量は指令された値が入力され、G91 モードで指令した場合、オフセット量は、前のデータに加算されます。

※工具補正機能を使用した実際のプログラム例は第 6 章応用編 P.99 「工具補正プログラム例」を参照してください。

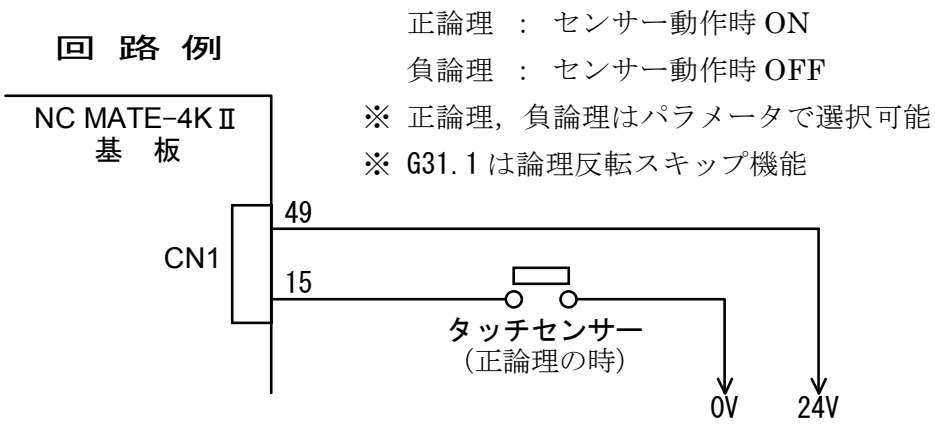
5・6　スキップ機能*1

パラメータ設定により、G31 で指令された移動ブロックを、スキップ信号（コネクター CN1 の 15 番ピン）により中断させ、次のブロックへ進ませることができます。
主な用途として、工具やワークの位置が段取りにより変化する場合、スキップ機能とタッチセンサーで位置を補正して加工したり，加工精度を測定することができます。

Gコード	内容
G31	スキップ機能
G31.1	論理反転スキップ機能

【プログラム例】

G91 G31 X20 F50・・・スキップ信号が入力された場合 X 軸の移動を中断し、次のブロックへ進む
G31.1 X-0.1 F1・・・・反転スキップ信号検出方式により更に高精度な位置検出が可能
　:
G90 G00 X0 Y0・・・・・スタート点へ戻る



相対座標のファイル記録（G132,G133）指令と組み合わせることにより、タッチセンサーで測定した相対座標を Data.txt ファイルに追加記録できるため、自動計測システムやデジタイザーなどに応用できます。

【プログラム例】

G90 G00 X100 Y200 Z0 ..... 測定個所に接近
G91 G01 G31 Z10 F30 ..... センサーが ON なら次のブロックへ
G91 G31.1 Z-10 F5 ..... センサーが OFF なら次のブロックへ
G133 ..... ファイル（Data.txt）の内容をクリヤー
G132 ..... 現在位置をファイルに記録
G90 G00 X110 Y200 Z0 ..... 次の測定個所に接近
G91 G01 G31 Y-10 F30 ..... タッチセンサーにて計測し、次のブロックへ
G91 G31.1 Y10 F5 ..... タッチセンサーが OFF なら次のブロックへ
G132 ..... 現在位置をファイルに追加記録
:

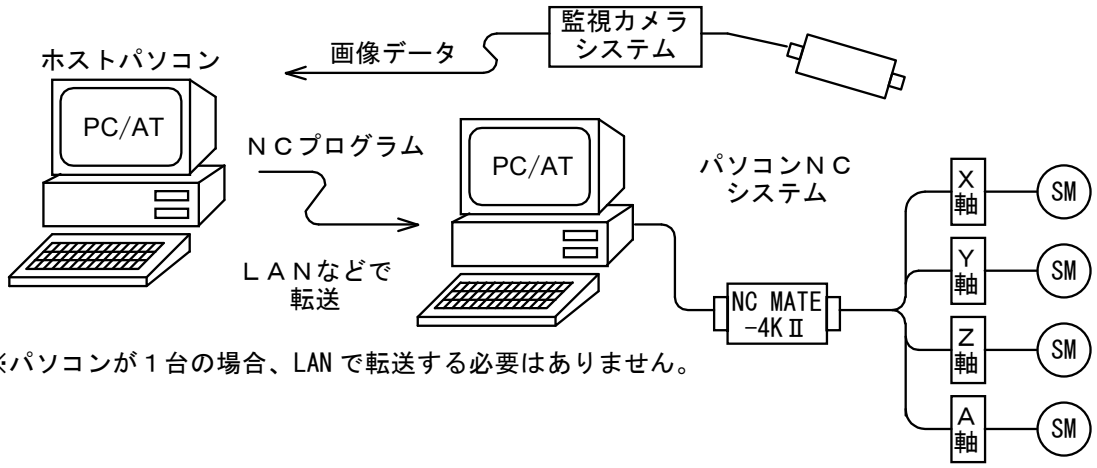
*1. NC MATE-4K Ⅱ BASIC では、使用できません。

５・７　遠隔制御運転機能*1

ホスト側で作成したデータを LAN などで結合し、指定共有ドライブ内に NC プログラムファイル（ファイル名 =DNC.ncm）を転送することにより、自動運転を起動できます。

Gコード	内容
G300	遠隔制御有効指令

DNC.ncm はサブプログラムやマクロ文，工具径補正やミラーイメージ，座標回転，スケーリング，IJK → R などの変換を含まない NC プログラムであることが必要です。
この機能を有効にするには、パラメータにドライブ名 232~239 を設定して下さい。
（232=C, 233=D, 234=E, 235=F, 236=G, 237=H, 238=I, 239=J）
DNC.ncm を転送するタイミングは、旧 DNC.ncm ファイルが無い時に転送して下さい。
読み込まれた DNC.ncm ファイルは、G300 指令で削除され、次の遠隔制御が有効となります。
※遠隔制御が ON で、運転準備完了または自動運転終了の時だけ遠隔運転が起動できます。



※パソコンが１台の場合、LAN で転送する必要はありません。

※遠隔制御運転機能を使用した実際のプログラム例は**第 6 章応用編 P.103 「遠隔制御運転プログラム例」**を参照してください。

*1. NC MATE-4K Ⅱ BASIC では、使用できません。

5・8　ティーチング機能*1

パラメータ設定とティーチングボタンオンにし、手動操作（早送り，粗送り，ハンドルなど）で軸移動を行い、記録したい位置で「スタート」をクリックするか、または外付の自動運転起動押ボタンを押すことにより、相対座標値を Data.Txt ファイルに追加記録できます。
また、相対座標のファイル記録指令 G132 によっても追加記録ができます。
このファイルをメモ帳などで編集（タイトルや G90，G00，M コードなど追加）し、名前を付けて保存することにより、ロボットなどの動作プログラムが容易に作成できます。

【Data.Txt 例】

```
X10.541 Y30.123 Z12.350 A1.000 .....記録位置１
X10.541 Y30.123 Z14.241 A2.000 .....記録位置２
X14.652 Y32.315 Z14.241 A3.000 .....記録位置３
          :                    :
X0.000 Y0.000 Z0.000 A0.000 .....スタート位置
```

【Data.Txt ファイルの編集例】

```
（ﾛﾎﾞｯﾄ A ﾌﾟﾛｸﾞﾗﾑ.....タイトルを付ける
 G90 G00.....アブソと早送りの指定をする
X10.541 Y30.123 Z12.350 A1.000 .....記録位置１
M10.....ハンドリング（把む）
X10.541 Y30.123 Z14.241 A2.000 .....記録位置２
X14.652 Y32.315 Z14.241 A3.000 .....記録位置３
M20.....ハンドリング（放す）
 :
X0.000 Y0.000 Z0.000 A0.000 .....スタート位置に戻る
/M99.....ブロックスキップ OFF の時、繰り返し
M02.....プログラム終了
```

Data.Txt の格納ドライブはパラメータ［144］で指定できます。
(0 ＝カレント，232=C，233=D, 234=E, 235=F, 236=G, 237=H, 238=I, 239=J)
記録軸の選択はパラメータ［143］で指定できます。
X 軸を記録する場合は 1 を加算して下さい。
Y 軸を記録する場合は 2 を加算して下さい。
Z 軸を記録する場合は 4 を加算して下さい。
A 軸を記録する場合は 8 を加算して下さい。
B 軸を記録する場合は 16 を加算して下さい。
X 軸，Z 軸を選択する場合は 1 ＋ 4 ＝ 5 を［143］に設定して下さい。
B 軸のデータは入力データ（IN0~IN3）の値を記録します。
記録座標はパラメータ［142］で変更できます。(0= 相対，1= 絶対，2= 機械座標)

スキップ機能と組み合わせる事により、モデルなどをタッチセンサーで自動的に計測して数値化する自動デジタイザーにも応用できます。

*1. NC MATE-4K Ⅱ BASIC では、使用できません。

5・9　ヘリカル補間機能*1

パラメータの変更により、準備機能（G02,G03）を用いてヘリカル補間ができます。
ヘリカル補間とは、円弧補間と同期して動く直線軸を指定することにより、螺旋状に移動するものです。
直線軸とは、G17(XY 平面指定）,G18(ZX 平面指定）,G19(YZ 平面指定）各々の平面と直交する軸、またはＡ軸とします。

G17 平面の時はＺ軸Ａ軸が直線軸で、G18 平面の時はＹ軸Ａ軸が直線軸、G19 平面の時はＸ軸Ａ軸が直線軸となります。

G17 G02 X100 Y100 Z100 R100 F100
　　円弧　直線

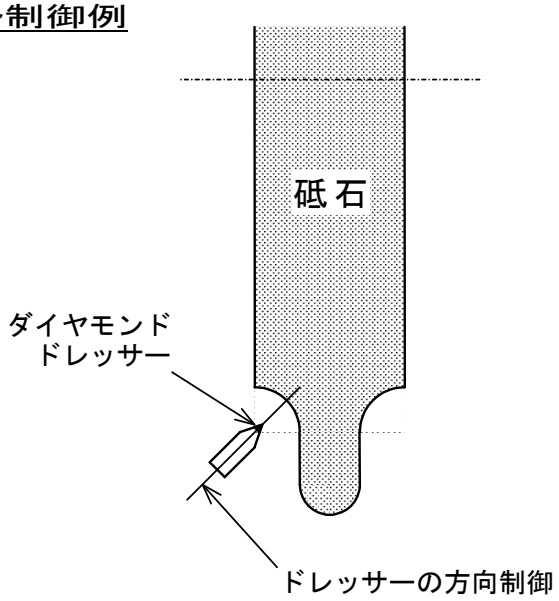
G18 G02 Z100 X-100 R-100 Y10A1
　　円弧　　直線

直線軸は、円弧補間に直線移動を付加しただけのものですから、送り速度は円弧の長さが基準になります。
従って、直線軸の送り速度は
Ｆ（直）＝Ｆ×（直線の長さ）／（円弧の長さ）

> ⚠ **円弧に対して直線が長い場合は、直線軸の送り速度が過大にならないように注意して下さい。**

【備考】ヘリカル補間は、上記螺旋状に補間する目的で作られた機能ですが、工具やトーチの姿勢制御にも使用できます。

姿勢制御例



*1.NC MATE-4K Ⅱ BASIC では、使用できません。

5・10　ピッチ誤差補正機能*1

パラメータの変更と補正データ作成により、各軸毎にピッチ誤差の補正ができます。

5・10・1　ピッチ誤差補正機能の概要

ピッチ誤差を各軸毎に最小設定単位で補正することができます。この機能は、各軸，リファレンス点復帰後に有効となります。
データはリファレンス点を基点としていますから、任意原点の場合は注意してください。

5・10・2　ピッチ誤差補正機能の仕様

表 5-2: ピッチ誤差補正機能の仕様

項目	仕様
補正可能な軸	全軸
補正点数	＋側　1000 点　／　－側　1000 点　　合計 2000 点
補正量範囲と補正間隔	0 ～ ±999 パルス 補正間隔（10mm 標準，パラメータ 106 番＝ 1 で 1mm）
補正方式	アブソリュート方式

5・10・3　ピッチ誤差補正データ作成方法

メモ帳でファイル名 PitchW.ncm を開き、データを下記要領で作成して下さい。
X 軸＋側は X の次に補正データ（0.001mm の時は 1）　と記載
Y 軸＋側は Y の次に補正データ（-0.001mm の時は -1）　〃
Z 軸＋側は Z の次に補正データ（0.002mm の時は 2）　〃
A 軸＋側は A の次に補正データ（-0.002mm の時は -2）　〃
X 軸－側は U の次に補正データ（0.001mm の時は 1）　〃
Y 軸－側は V の次に補正データ（-0.001mm の時は -1）　〃
Z 軸－側は W の次に補正データ（0.002mm の時は 2）　〃
A 軸－側は B の次に補正データ（-0.002mm の時は -2）　〃

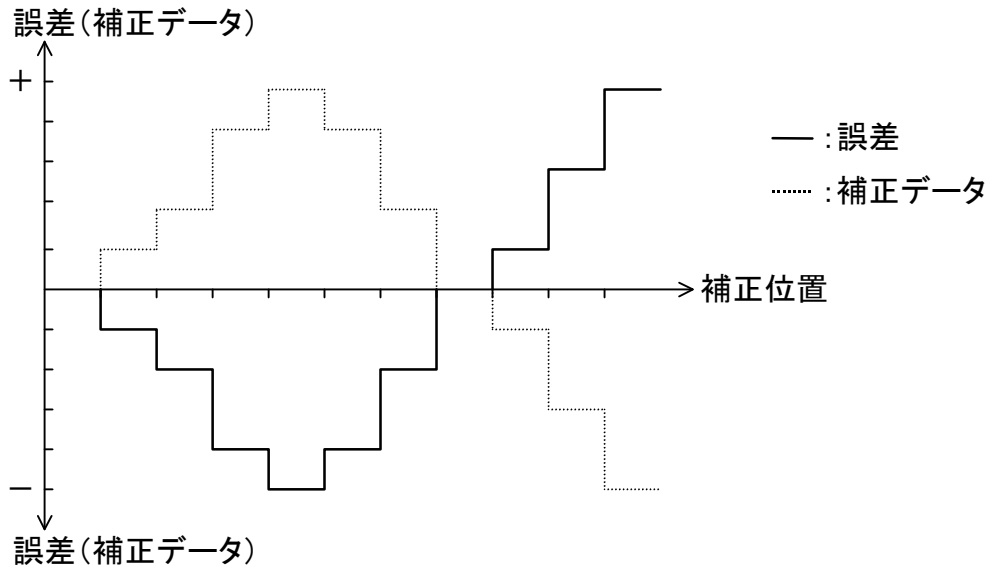
【ピッチ誤差補正データ作成例】（補正間隔 10mm の場合）
X1Y-1Z2A-2U1V-1W2B-2 .... ファレンス点より ±10 ～ 19.999mm 間の補正データ
X2Y-2Z2A-2U1V-2W3B-3 .... ファレンス点より ±20 ～ 29.999mm 間の補正データ
X2Y-1Z1U2V-1W4 ..... 使用しない軸は記載しなくても良い
　：　（使用しない軸のデータは 0 となります）
X1Y2W-2 ..... 範囲外のデータは記載しなくても良い
　：　（記載しない場合のデータは 0 となります）
100X3Y5 ..... 行の目印（100）を記載しても良い
　：
1000Y8 ..... 最大 1000 行（±10m）まで記載可能

*1. NC MATE-4K Ⅱ BASIC では、使用できません。

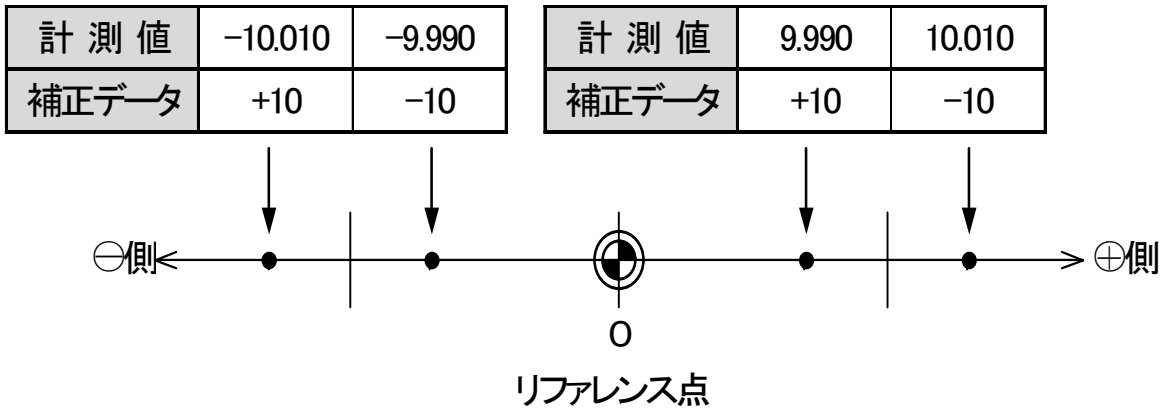
5・10・4　ピッチ誤差補正データ

補正データは、設定単位に応じて 0 ～ ±999 まで設定できます。
10mm 間隔で設定単位 0.001 の場合の例を示します。

No.	補正位置	計 測 値	ピッチ誤差	補正データ
1	10.00	9.999	−0.001	1
2	20.00	19.998	−0.002	2
3	30.00	29.996	−0.004	4
4	40.00	39.995	−0.005	5
5	50.00	49.996	−0.004	4
6	60.00	59.998	−0.002	2
7	70.00	70.000	0.000	0
8	80.00	80.001	0.001	−1
9	90.00	90.003	0.003	−3
10	100.00	100.005	0.005	−5



＋側，－側 で補正データの符号は下記の通りです。

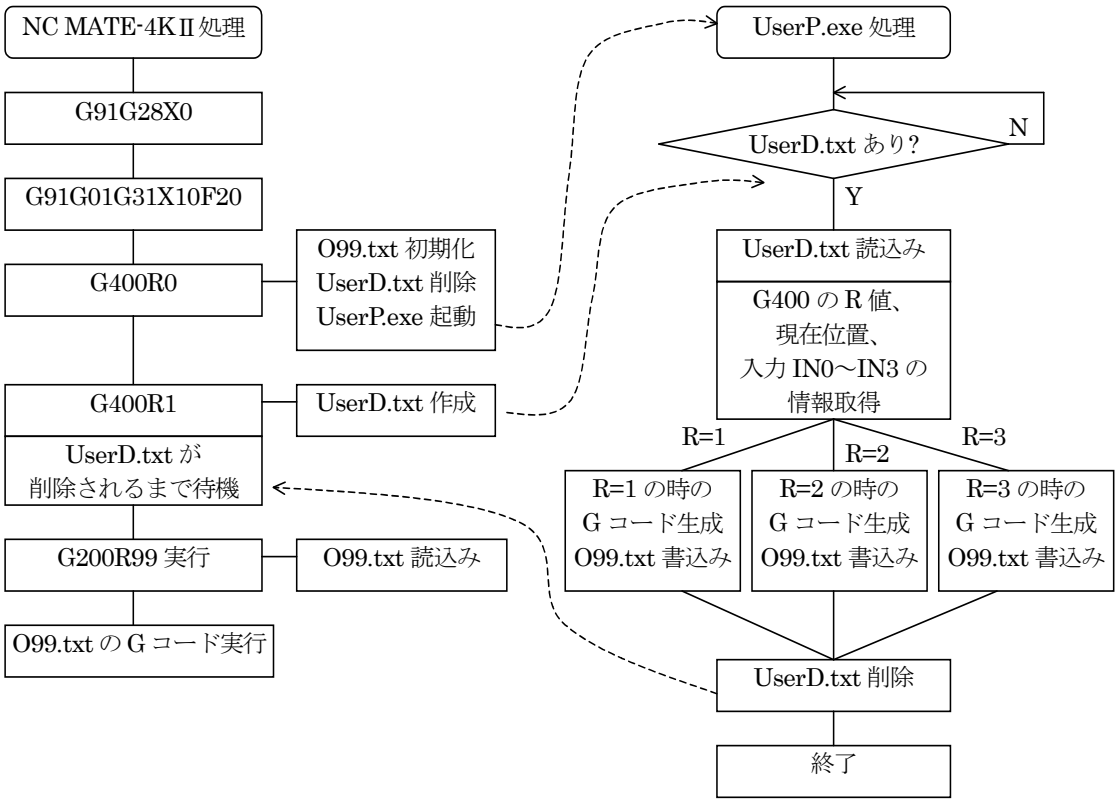


5・11　ユーザーマクロリンク機能*1

パラメータの変更により、ユーザーマクロリンク機能が使用できます。
この機能により、Delphi や C，BASIC コンパイラーなどを使用して開発されたユーザーのプログラムとリンクすることができ、変数や演算機能を必要とする高度な制御が可能となります。

G コード / ファイル	内容
G400	ユーザーマクロリンク指令
G400R0	ユーザーの作成した実行ファイル UserP.exe を起動します。 UserP.exe を起動する前に O99.txt は初期化（空のプログラムに書換え）され、 UserD.txt は削除されます。
G400R1 ～ 999	UserD.txt に R の値、各軸の現在位置、入力 IN0 ～ IN3 の情報を書き込みます。 その後、UserD.txt が削除されるまで待機します。
UserP.exe	ユーザーが作成した実行ファイル。この実行ファイルの中で UserD.txt を読み込み、 必要な G コードを生成し、O99.txt に書き込み、UserD.txt を削除します。
UserD.txt	NC 制御ソフトから UserP.exe へ、入力情報、現在値情報を渡すためのファイル
O99.txt	UserP.exe で生成される G コードのファイル

＜使用例＞
G91G28X0　　X 軸原点復帰
G91G01G31X10F20　スキップ指令でX 軸機械座標を検出
G400R0　　ユーザープログラムを立上げ
G400R1　　データ番号 1 のユーザプログラムを実行する指令
G200R99　　データ番号 1 で作成するユーザ作成プログラム 1 の実行指令
M02



*1. NC MATE-4K Ⅱ BASIC では、使用できません。

５・１１・１　プログラムの流れと注意事項

```
098.Txt には下記のプログラムをしてマクロを呼び出す。
（ユーザーマクロテスト）
G91G28X0            (X 軸原点復帰)
G91G01G31X10F20     (スキップ指令で X 軸機械座標を検出)
G400R0              (ユーザープログラムを立上げ)
G400R1              (データ番号１のユーザプログラムを実行する指令)
G200R99             (データ番号１で作成するユーザ作成プログラム１の実行指令)
M02
```

◆プログラムの流れは、098.Txt を NC MATE-4K Ⅱで実行すると、X 軸原点復帰を実行し、次にスキップ指令で X 軸機械座標を検出した後、G400R0 で旧 UserD.Txt を削除し、ユーザープログラム UserP.Exe を自動的に立ち上げ、二重起動防止チェックをします。
◆次の指令 G400R1 でデータ番号１と X ～ A 軸の座標と IN0 ～ 3 のデータを UserD.Txt に記録した後、UserP.Exe に制御を移し、データ番号１の時の処理を実行させます。
◆データ番号１の時の処理は、099.Txt に
G90G53G01X0F100
G91G01G31X10F20
G400R2G200R99
M02
を書き込み後、UserD.Txt を削除する事により NC MATE-4K Ⅱに制御を移します。
◆ NC MATE-4K Ⅱで G200R99 を実行すると、099.Txt に記載された G90G53G01X0F100 を実行します。
◆次に G91G01G31X10F20 のスキップ指令でポイント２の X 軸機械座標を検出して、次の指令 G400R2　でデータ番号２と X ～ A 軸座標と IN0 ～ 3 のデータを UserD.Txt に記録した後、UserP.Exe に制御を移し、データ番号２の時の処理を行います。
◆データ番号２の時の処理は、099.Txt にポイント１と２の平均位置に移動させる指令 G90G53G00X'+stra を書き込み後、UserD.Txt を削除する事により、NC MATE-4K Ⅱに制御を移し、NC MATE-4K Ⅱで G200R99 で実行することで 099.Txt に記載された G90G53G00X'+stra を実行します。
◆上記によりポイント１と２の平均位置に X 軸を位置決めする事が可能となります。この後、G200 指令（ファイル連結指令）で別のプログラムに連結することもできます。
◆データ番号２の処理では、二重起動を防止するため、UserP.Exe を Close させています。
◆安全のため、099.Txt は G400R0 ～ 9999 指令で NC MATE-4K Ⅱにより初期化されます。

※ユーザーマクロリンク機能を使用した実際のプログラム例は**第６章応用編 P.101 「ユーザーマクロ開発例」**を参照してください。

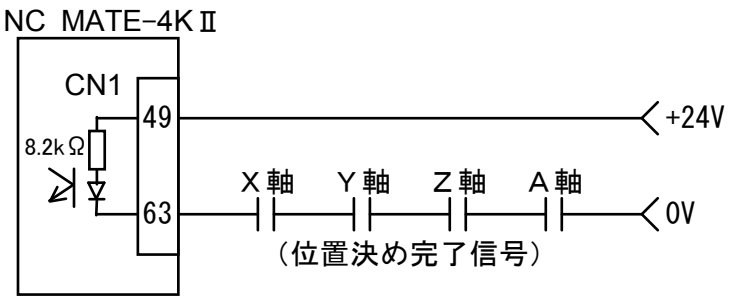
> ⚠ **・ユーザープログラム UserP.Exe は NC MATE-4K Ⅱと同じフォルダに追加して下さい。**
> **・UserP.Exe は二重起動しないようにする必要があります。**
> **・099.Txt は G400（ユーザーマクロリンク指令）R0 ～ 9999 指令で内容が変化します。**

5・12　イグザクトストップ機能

サーボのたまりパルスによるコーナ部の丸みを防ぐ為、プログラムのブロック間にパラメータで設定された時間を設ける機能で、G61 で有効、G64 でキャンセルされます。
停止時間は（パラメータで設定された値）×（0.001sec.）となります。
また、G09 で１ブロックだけ有効にすることもできます。
特別モードとして、パラメータを 10000 に設定すると、サーボドライバの位置決め完了信号を検出してから次のブロックへ進みます。

Ｇコード	内容
G09	イグザクトストップ（ワンショット）
G61	イグザクトストップモード有効
G64	イグザクトストップモードキャンセル

位置決め完了信号の接続例



※パラメータ＝ 10000 のとき、位置決め完了信号を確認します。）

5・13　外付押ボタン操作機能

パラメータの変更により、外付押ボタン操作方式に変更することができます。

この場合、モード選択の際は荒送りを選択してオーバーライド SW で JOG 速度の調整ができます。 速度の基準は、100%の値を荒 JOG のパラメータに設定しますと、10%変化する毎に約 1.6 倍の指数関数値になります。

外

荒 JOG のパラメータ［60～63］を 100 に設定した場合の例

オーバーライド（%）	0	10	20	30	40	50	60	70	80
送り速度（mm/min.）	0	1.5	2.3	3.7	6.0	9.5	15.3	24.1	39.1

オーバーライド（%）	90	※100	110	120	130	140	150
送り速度（mm/min.）	62.5	※100.0	160.0	256.0	410.0	655.0	1049.0

【注】送り速度が 50 以下の場合は設定値との差が大きくなり、2.9 以下は 0 となります。

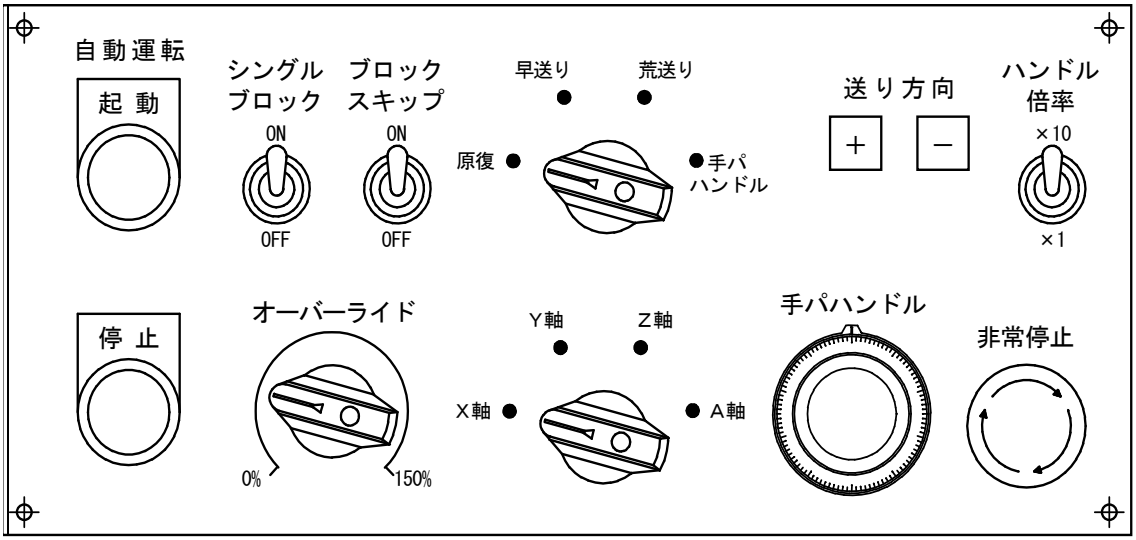
付押ボタン操作項目は次のとおりです。

名称	スイッチ仕様
自動運転「起動」	PB
自動運転「停止」	PB
シングルブロック「ON/OFF」	SS
ブロックスキップ「ON/OFF」	SS
移動モード選択	SS
軸選択	SS
＋方向	PB
－方向	PB
ハンドル倍率「×1/×10」	SS
オーバーライド「0 ～ 150%」	SS

５・13・１　操作方法およびレイアウト例

操作方法は、キーボードのファンクションキーや各々のモード選択用のキーと同様に、画面上の表示に従って必要なスイッチを操作します。
なお、操作に先立って、拡張モードの入力状態表示画面で、結線などを確実にチェックしてください。

外部操作盤レイアウト例



５・13・２　オーバーライドスイッチの速度倍率

モード選択で粗送りを選択して、オーバーライド SW で、送り速度の調整をすることができます。速度の基準は、100% の値を粗送りのパラメータに設定しますと、10% 変化するごとに、約 1.6 倍の指数関数値になります。

【粗送りのパラメータを 100 に設定した場合の例】

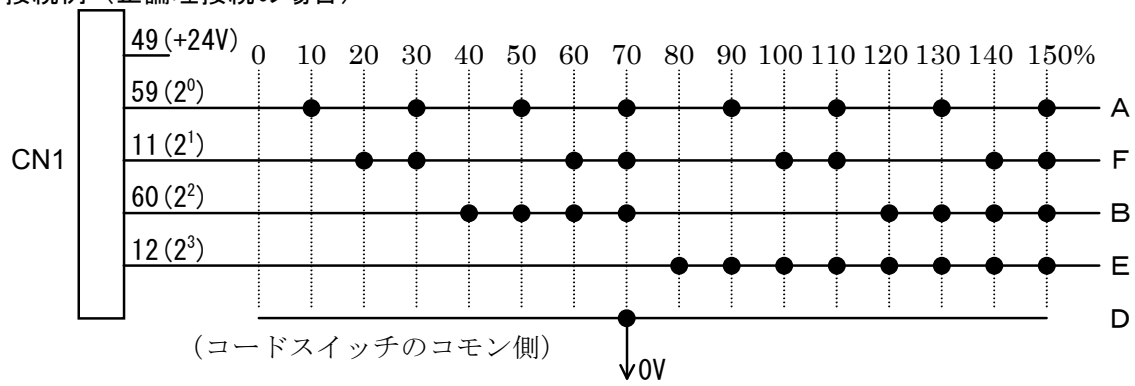
オーバーライド（%）	0	10	20	30	40	50	60	70
送り速度（mm/min.）	0	1.5	2.3	3.7	6.0	9.5	15.3	24.1

オーバーライド（%）	80	90	100	110	120	130	140	150
送り速度（mm/min.）	39.1	62.5	100	160.0	256.0	410.0	655.0	1049.0

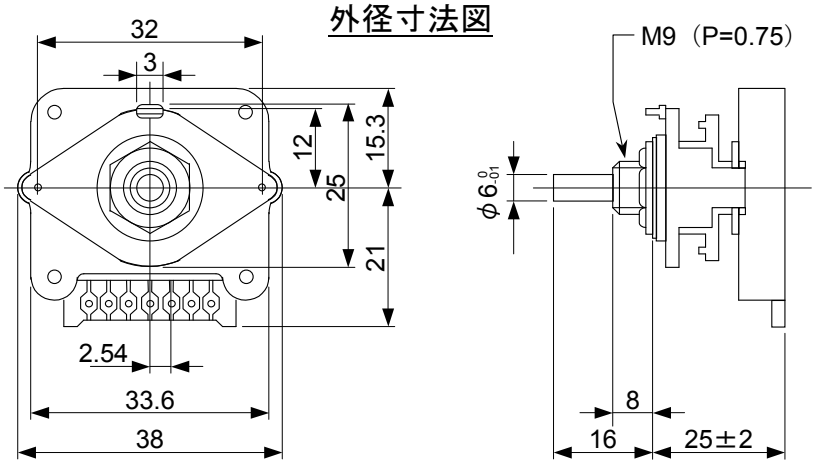
⚠ 送り速度が、50 以下の場合は、設定値との差が大きくなり、2.9 以下は 0 となります。

5・13・3　オーバーライド用コードスイッチの接続方法

接続例（正論理接続の場合）



外径寸法図



オーバーライド用コードスイッチ 東京測定器材　DPP01015J16R 相当 当社オプション品 : ε CS-DPP01015J16R ※取り付けの際には、最新情報をメーカーホームページでご確認ください。

５・１３・４　外付け押しボタン操作に関するパラメータについて

パラメータ番号［56］：外付け押しボタン機能のパラメータ値を変更することにより、外付け押しボタンを有効にすることができます。

設定値	機能
0	無効
30	有効（オーバーライドスイッチは正論理）
31	有効（オーバーライドスイッチは負論理）

【オーバーライドスイッチ設定と速度倍率】

パラメータ番号［56］の設定値が 30 の場合

	0V3	0V2	0V1	0V0
150%	ON	ON	ON	ON
140%	ON	ON	ON	OFF
130%	ON	ON	OFF	ON
120%	ON	ON	OFF	OFF
110%	ON	OFF	ON	ON
100%	ON	OFF	ON	OFF
90%	ON	OFF	OFF	ON
80%	ON	OFF	OFF	OFF
70%	OFF	ON	ON	ON
60%	OFF	ON	ON	OFF
50%	OFF	ON	OFF	ON
40%	OFF	ON	OFF	OFF
30%	OFF	OFF	ON	ON
20%	OFF	OFF	ON	OFF
10%	OFF	OFF	OFF	ON
0%	OFF	OFF	OFF	OFF

パラメータ番号［56］の設定値が 31 の場合

	0V3	0V2	0V1	0V0
150%	OFF	OFF	OFF	OFF
140%	OFF	OFF	OFF	ON
130%	OFF	OFF	ON	OFF
120%	OFF	OFF	ON	ON
110%	OFF	ON	OFF	OFF
100%	OFF	ON	OFF	ON
90%	OFF	ON	ON	OFF
80%	OFF	ON	ON	ON
70%	ON	OFF	OFF	OFF
60%	ON	OFF	OFF	ON
50%	ON	OFF	ON	OFF
40%	ON	OFF	ON	ON
30%	ON	ON	OFF	OFF
20%	ON	ON	OFF	ON
10%	ON	ON	ON	OFF
0%	ON	ON	ON	ON

※ 0V3 ～ 0V0 は、NC MATE-4K Ⅱへの入力信号です。

このオーバーライドスイッチは、外付け押しボタン操作で「粗送り」を選択した時のみ有効です。「早送り」には影響しません。

5・13・4-1　早送りオーバーライドについて

パラメータ番号 [19]：早送りオーバーライドのパラメータ値を変更することにより、早送りオーバーライド機能を有効にすることができます。

設定値	機能
0	有効
1	無効

①このパラメータは、G00 指令にのみ影響を与えます。
②パソコンから操作するとき（外付け押しボタン操作が無効の時）、送り方向＋ボタン及び－ボタンは、すべてのモード選択において、オーバーライド速度に影響されません。

5・13・4-2　マウス正逆寸時に外付け正逆も有効について

パラメータ番号 [102]：マウス正逆寸時に外付け正逆も有効のパラメータ値を変更することにより、マウス正逆寸時に外付け正逆も有効の機能を有効にすることができます。

設定値	機能	
0	無効	外付け押しボタン操作が有効の時：外付け押しボタン操作有効 外付け押しボタン操作が無効の時：外付け押しボタン操作無効
1	有効	外付け押しボタン操作が有効の時：外付け押しボタン操作有効 外付け押しボタン操作が無効の時：外付け押しボタン操作有効

このパラメータを有効にすると、外付け押しボタン操作が無効に設定されている場合でも、**外付け押しボタンで操作**することが可能となります。
このとき、パソコンからの操作も外付け押しボタンからの操作も有効となります。
パソコン側と、外付け押しボタン側と同時に押した場合は、先に押した方の操作が有効となります。（先取り優先）

5・14　スケーリング機能

パラメータの変更により、加工プログラムで指令した形状の縮小，拡大（スケーリング）ができます。

G51 X＿ Y＿ Z＿ P＿

X, Y, Z ：スケーリング中心座標値（対象軸）
P　　：倍率（0.001 ～ 1000）

この指令により、以降の移動指令はPで指定された倍率だけ中心点よりスケーリングした値に変換されます。スケーリングキャンセルは、単独でG50を指令してください。

Gコード	機能
G51	スケーリング
G50	スケーリングキャンセル

5・14・1　スケーリングの注意事項

◆スケーリングは、工具径補正と同様にプログラム読み込み時に変換を行いますので、実行前に軌跡チェックなどができます。
◆スケーリング開始時に中心座標値を設定した軸だけがスケーリングされますので、各軸毎の有効／無効の選択が容易にできます。
◆スケーリング中心座標を指定しなかった場合は、G51を指令した位置が中心になり、全軸ともスケーリングを行います。
◆スケーリング倍率Pを指定しなければ［**スケーリング倍率アラーム**］となります。
◆正確に変換できない場合は、手入力にて修正をしてください。

※スケーリング機能を使用した実際のプログラム例は**第６章応用編 P.104 「スケーリングプログラム例」**を参照してください。

5・15　プログラマブルミラーイメージ機能

パラメータの変更により、加工プログラムで指令した形状を各軸ごとにミラーイメージ（反転）をかけることができます。

G51.1 X＿ Y＿ Z＿

X, Y, Z ：ミラーイメージをかける軸および座標

この指令により、以降の移動指令は指定された各軸の位置を中心に反転した値に変換されます。
ミラーイメージのキャンセルは G50.1 を単独で指令してください。

Gコード	機能
G51.1	プログラマブルミラーイメージ
G50.1	プログラマブルミラーイメージキャンセル

5・15・1　プログラマブルミラーイメージの注意事項

◆プログラマブルミラーイメージも工具径補正と同様にプログラム読み込み時に変換を行いますので、実行前に軌跡チェックなどができます。
◆ G51.1 で軸を指定しなかった場合は、ミラーイメージはどの軸にもかかりません。
◆指定平面の１軸のみにミラーイメージをかけた場合、以下の指令は次のようになります。
◆円弧指令→時計回りと反時計回りが逆になります。
◆工具径補正 →右オフセットと左オフセットが逆になります。
◆正確に変換できない場合は手入力にて修正をしてください。

※プログラマブルミラーイメージ機能を使用した実際のプログラム例は**第 6 章応用編 P. 105 「プログラマブルミラーイメージプログラム例」**を参照してください。

5・16　座標回転機能

パラメータの変更により、加工プログラムで指令した形状を回転させることができます。

G68 α＿ β＿ R＿

α、β　：回転中心の座標値
　　G17,G18,G19 で選択された平面内の２軸（Ｘ－Ｙ，Ｚ－Ｘ，Ｙ－Ｚ）を指定します。
R　　：回転角度（0.001 ～ ±360.0deg）
　　反時計方向，左回りが ＋ です。
　　アブソリュート値，インクリメンタル値が使用できます。

この指令により、以降の移動指令は α，β を中心として R で指定された角度だけ回転した値に変換されます。座標回転キャンセルは単独で G69 を指令してください。

Gコード	機能
G68	座標回転
G69	座標回転キャンセル

5・16・1　座標回転の注意事項

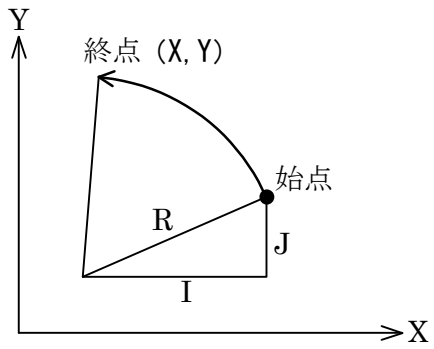
◆座標回転も工具径補正と同様にプログラム読み込み時に変換を行いますので、実行前に軌跡チェックなどができます。
◆回転中心を指令しなかった場合は、G68 を指令した位置が回転中心になります。
◆回転角度を指令しなかった場合は座標回転は行われません。
◆インクリメンタル値で回転角度を指令して連続して回転させたい場合は、途中で G69 でキャンセルせずに G68 を続けて指令してください。
◆正確に変換できない場合は、手入力にて修正をしてください。

※座標回転機能を使用した実際のプログラム例は**第 6 章応用編 P. 106 「座標回転プログラム例」**を参照してください。

5・17　円弧補間 IJK 指定

パラメータの変更により、円弧補間を半径 R ではなく、中心までの距離（I，J，K の内 2 軸）を使って表示することができます。

```
G17 G02 X__ Y__ I__ J__
G18 G03 Z__ X__ K__ I__
G19 G02 Y__ Z__ J__ K__
```



IJK の数値は G90,G91 にかかわらず、常にインクリメンタル値とし、始点からの方向に応じて符号を付けてください。

5・17・1　IJK 表示の注意事項

◆ IJK 指定は、プログラム読み込み時に R 指定に変換します。従って、実行時には IJK 表示は消えてしまいます。

◆ I0,J0,K0 は省略できない場合があります。

◆移動量 0 で 360゜の円弧（全円）を指令した場合は、180゜ずつの円弧（半円）に分割されます。

```
G91 G02 X0 Y0 I50 → G02 X100 Y0 R50
                    G02 X-100 Y0 R50
```

◆ IJK の値は正確にプログラムする必要があります。IJK の値が正確でない場合は、円弧の精度が悪くなります。

◆ IJK の値が小さくて終点の座標に届かない場合は円弧設定エラーとなりますので、エラーが発生しない値に修正してください。また、CAD/CAM によっては $\sqrt{(I)^2+(J)^2+(K)^2}$=R の値を少なく計算する場合があり、R の値が終点の座標に届かない時があります。

◆平面指定（G17,18,19）とインクリメンタル・アブソリュート指定（G90,91）は、必ずプログラム中に記載されている必要があります。

◆正確に変換できない場合は、手入力にて修正をお願いします。

5・18　ファイル連結機能（大容量ファイル運転）

パラメータの変更により、ファイルサイズの限界（399999 ステップ，10MB）を越えるような長いプログラムが必要な場合は、ファイルを分割した後、実行時に連続して動作させることができます。

G200 R＿
　R ：ファイル No.（1 ～ 9999）

ただし、工具長補正や実行前にコード変換を伴う工具径補正や各種補正機能などは、オフセットモードのままではファイル連結できませんので、必ずキャンセルモードで指令してください。
続けて補正をかけたい場合には、ファイルの終了時に一旦キャンセルして次のファイルの実行時に再び補正をかけてください。
また、長いプログラムを分割する場合は、工具を切削点から離すようにして終了させ、連結する時に工具を切削点に戻すようにプログラムを修正してください。
また、連結される側のプログラムは、独立したプログラムとして実行できるように、必要な G コードなどは必ず再度プログラムしてください。

G コード	機能
G200	ファイル連結指令

【プログラム例】

```
（Test of G200）..... タイトル
G90 G10 P5 R5.001 ..... No.5 番の工具径補正量設定
G91 G28 X0 Y0 Z0 ..... 自動原点復帰
G200 R10..... プログラム  O10 を読み込み実行
M02
```

> **200 指令の次のステップには M02 指令を必ずプログラムしてください。**
> **また、G200 指令は 1 指令だけプログラムが可能で、他の指令と併用できません。**

G200 指令は MDI で実行できますから、各種変換を必要とするプログラムの高速読込と実行ができます。
また、遠隔制御運転でホストにて作成したデータに G200 指令をプログラムして転送しますと、工具径補正など各種変換を行い、自動運転を実行することが可能です。

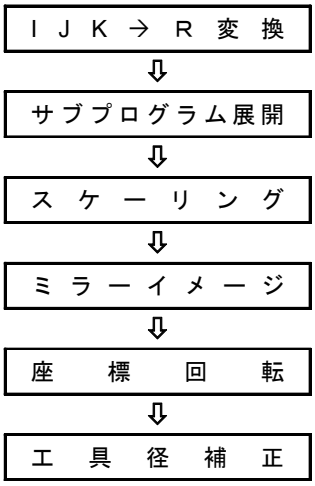
5・19　ファイル変換方式について

既に各々の項で述べているように、工具径補正を始め、IJK 指定スケーリング，プログラマブルミラーイメージ，座標回転などは、全てプログラム読み込み時に補正計算を行うファイル変換方式を採用しています。

これらの機能を単独で使用する場合は特に問題ありませんが、数種類の機能を組み合わせて使用する場合には次の点に考慮してください。

5・19・1　変換順序

ソースプログラムを変換する順序は次の通りです。



5・19・2　座標回転とミラーイメージの組み合わせ

座標回転とミラーイメージを組み合わせて使用する際には、プログラムの構成に関係なく、ミラーイメージ計算を先に行います。従って、座標回転させた結果を反転させたいような場合には、次のような操作を行ってください。

1. まず、座標回転だけを計算させるために、ミラーイメージの開始コード G51 のブロックに“(”をつけてコメントにして終了させてください。
2. その座標回転された結果を別の No. に登録します。
3. 次に、その登録されたプログラムの G51 ･･･ のブロックの“(”を外して、ミラーイメージ計算をさせてください。

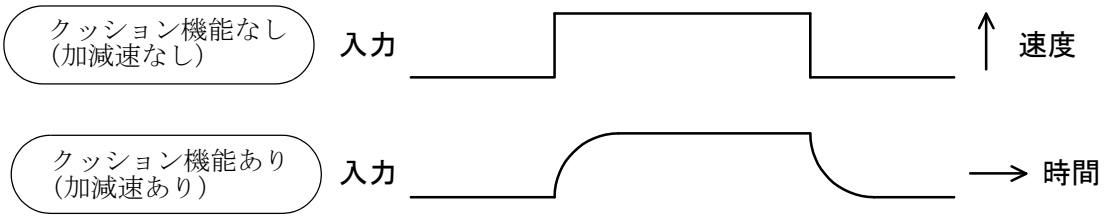
これで、座標回転後の軌跡を反転させることができます。

※座標回転とミラーイメージの組み合わせた実際のプログラム例は**第 6 章応用編 P. 107 「組み合わせプログラム例」**を参照してください。

5・20　クッション機能（なめらか制御）

クッション機能は、入力されたパルス周波数（モータスピード）に対し指数加減速を行い、サーボモータやパルスモータの加減速時のショックを少なくすることができます。
パラメータ番号 115 ～ 118 で加減速時定数を設定することができます。

入出力速度の関係は下図のようになります。



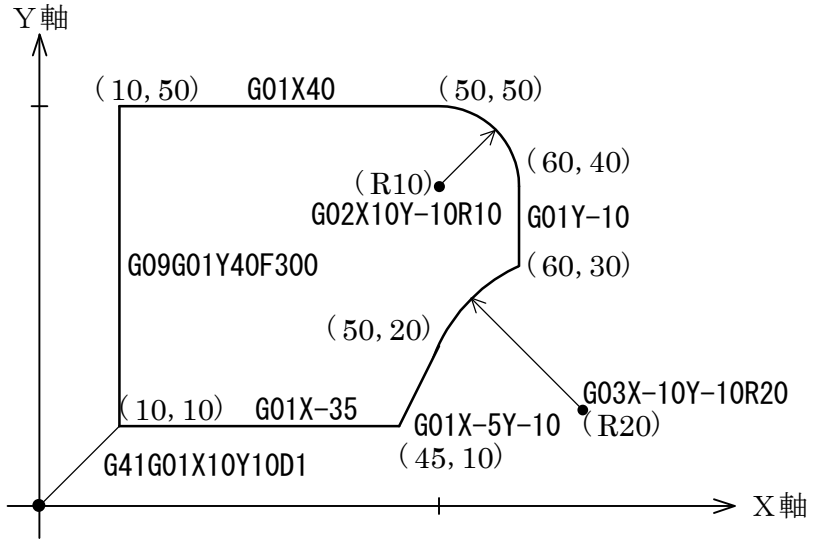
本機能は、主にクッション機能をもたないサーボドライバーで、2m/min. 以上の切削送り（G01）を必要とする場合や、パルスモータを自起動周波数以上で使用する場合に必要です。

円弧補間を高精度で行う場合は、10ms ～ 40ms の間でのご使用を推奨いたします。

クッション機能加減速時定数を 0 に設定した場合は、クッション機能は無効になります。

第 6 章　応用編

6・1　NC 加工プログラムの作成例



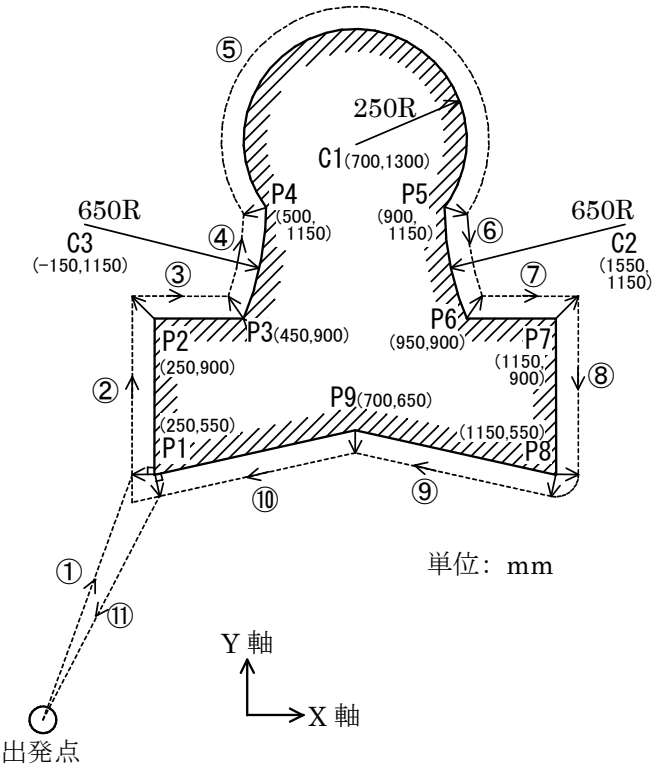
【ＮＣ 加工プログラム作成例】

```
(SAMPLE PROGRAM) ..... (タイトル)
G91 G17 G01 ..... (初期設定, インクリメンタル, X－Y 平面)
G27 X0 Y0 Z0 ..... (スタート点 チェック)
G43 H1 F50..... (工具長補正)
M98 P100 ..... (サブプログラム呼び出し)
G41 G01 X10 Y10 D1 ..... (工具径補正 スタート)
G04 X1 ..... (時間待ち 1秒)
G09 G01 Y40 F300..... (イクザクトストップで Y 軸 40mm)
G01 X40 ..... (X 軸 40mm　速度 300mm/min. )
G02 X10 Y-10 R10..... (時計方向円弧　R10mm)
G01 Y-10 ..... (Y 軸 -10mm)
G03 X-10 Y-10 R20 ..... (反時計方向円弧　R20mm)
G01 X-5 Y-10 ..... (X 軸 -5mm　Y 軸 -10mm)
G01 X-35 ..... (X 軸 -35mm)
G40 G00 X-10 Y-10 ..... (工具径補正キャンセル)
G49..... (工具長補正キャンセル)
M30..... (終 了)

O100..... (サブプログラム　O100)
G01 Z-30 F100 ..... (Z 軸 -30mm　速度 100mm/min. )
/M00..... (ブロック スキップ　オフの場合一時停止)
G00 Z30 ..... (Z 軸 30mm 早送り速度)
M99..... (サブプログラム終了)
```

6・2　工具補正プログラム例

【例１】

```
G92 X0 Y0 Z0 ;
G90 G17 G01 G43 H01 F150 ;
G41 D01 X250.0 Y550.0 ;……①
Y900.0 ;  ……②
X450.0 ;  ……③
G03 X500.0 Y1150.0 R650.0 ;……④
G02 X900.0 R-250.0 ; ……⑤
G03 X950.0 Y900.0 R650.0 ; ……⑥
G01 X1150.0 ;……⑦
Y550.0 ;  ……⑧
X700.0 Y650.0 ;……⑨
X250.0 Y550.0 ;……⑩
G40 X0 Y0 ;……⑪
G49 ;  ……⑫
M02 ;
```



上記のようにプログラムされたものを、
工具Ｒとして D01 に 10mm オフセットデータをセットした場合、工具補正計算後のプログラムは下記のようになります。

```
G92 X0 Y0 Z0 ;
G90 G17 G01 G43 H01 F150 ;……工具長補正
D01 X240 Y550 ; ……①'
X240 Y910 ; ……②'
X443.296 Y910 ; ……③'
G03 X489.992 Y1146.714 R640 ; …④'
G02 X910.008 Y1146.714 R-260 ;…⑤'
G03 X956.704 Y910 R640 ; …… ⑥'          工具径補正
G01 X1160 Y910 ;  ……⑦'
X1160 Y540 ;  ……⑧'
X1157.593 Y538.069 ;
X700 Y639.756 ; ……⑨'
X252.169 Y540.238 ; ……⑩'
X0 Y0 ;  …… ……⑪'
G49 ; ……工具長補正キャンセル
M02 ;
```

【例２】サブプログラムを使用した例（D01 ＝ 10mm，D02 ＝ 9.5mm）

(補正計算前)

```
G92 X0 Y0 Z0 ;
D01 H01 F150 ;
M98 P100 L1 ;
D02 H02 F100 ;
M98 P100 L1 ;
M02 ;
O100 ;
G90 G17 G01 G43 ;
G41 X250.0 Y550.0 ;
Y900.0 ;
X450.0 ; G01 X1160 Y910 ;
G03 X500.0 Y1150.0 R650.0 ;
G02 X900.0 R-250.0 ;
G03 X950.0 Y900.0 R650.0 ;
G01 X1150.0 ;
Y550.0 ;
X700.0 Y650.0 ;
X250.0 Y550.0 ;
G40 X0 Y0 ;
G49 ;
M99 ;
```

(補正計算後)

```
 G92 X0 Y0 Z0 ;
 D01 H01 F150 ;
 (Start SUB P100 L1 Nest1)
 G90 G17 G01 G43 ;
X240 Y550 ;
X240 Y910 ;
X443.296 Y910 ;
G03 X489.992 Y1146.714 R640 ;
G02 X910.008 Y1146.714 R-260 ;
G03 X956.704 Y910 R640 ;

X1160 Y540 ;
X1157.593 Y538.069 ;
X700 Y639.756 ;
X252.169 Y540.238 ;
X0 Y0 ;
G49 ;
(End SUB P100)
D02 H02 F100 ;
(Start SUB P100 L1 Nest1)
G90 G17 G01 G43 ;
X240.500 Y550 ;
X240.500 Y909.500 ;
X443.633 Y909.500 ;
G03 X490.492 Y1146.876 R640.500 ;
G02 X909.508 Y1146.876 R-259.500 ;
G03 X956.367 Y909.500 R640.500 ;
G01 X1159.500 Y909.500 ;
X1159.500 Y540.500 ;
X1157.213 Y538.665 ;
X700 Y640.268 ;
X252.061 Y540.726 ;
X0 Y0 ;
G49 ;
(End SUB P100) ;
M02 ;
```

6・3　ユーザーマクロ開発例

※Delphi や C を使用し、ユーザーにて作成

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  StdCtrls, ExtCtrls;

type
  TForm1 = class(TForm)
    Timer1: TTimer;
    Label1: TLabel;
    procedure Timer1Timer(Sender: TObject);
    procedure FormShow(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
  private
    { Private 宣言 }
  public
    { Public 宣言 }
  end;

var
  Form1: TForm1;

implementation

{$R *.DFM}

var
  UsrD:array[0..10] of string[100];   { unit 内で使用するグローバル変数の宣言}
  Xpoint1,Xpoint2:Double; Pas:String; Drunf:integer;

procedure TForm1.FormShow(Sender: TObject);      { UserP.Exe 立ち上げ時の処理 }
 var F:TextFile;                                    { ローカル変数を宣言する}
begin
  Pas:=ExtractFilePath(Application.ExeName);          { ファイルのパスを取得 }
  Drunf:=0;                                           { 二重起動フラグの初期化}
  if FileExists(Pas+'UserR.Txt')=True then Drunf:=1; { UserR.Txt が存在する場合}
  if Drunf=1 then
   begin
     ShowMessage('二重起動されているか又は、UserR.Txt が削除されていません。');
     Close;              { 二重起動の時はユーザプログラム UserP.Exe をクローズ }
   end;
  if Drunf=0 then                                    { UserR.Txt が存在しない時}
   begin
    AssignFile(F,Pas+'UserR.Txt');                   { UserR.Txt ファイル選択 }
    update;                                          { 情報をリフレッシュする }
    Rewrite(F);                                      { UserR.Txt ファイル初期化}
    Writeln(F,'二重起動防止チェックファイル');       { 第１行にタイトル書込み }
    CloseFile(F);                                    { UserR.Txt 作成とクローズ}
    Form1.Left:=0; Form1.Top:=0;                     { フォーム表示位置を設定 }
    Timer1.Interval:=50;           { 立ち上げ時無効のタイマーを 50ms に有効設定}
   end;
end;
```

```
procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin                    {正常終了の時二重起動防止チェック用 UserR.Txt を削除する}
 if Drunf=0 then DeleteFile(Pas+'UserR.Txt');
end;


procedure TForm1.Timer1Timer(Sender: TObject);       { タイマーが作動したら実行}
  var F:TextFile; stra:string;                       { ローカル変数を宣言する}
 begin
   AssignFile(F,Pas+'UserD.Txt');                    { UserD.Txt ファイルを選択}
   if FileExists(Pas+'UserD.Txt')=False then exit;  { ファイルがない時 }
    update; {AssignFile と reset 間に update;を入れないと立ち上げ時 I/O エラーになる}
    try reset(F); except exit end;       { reset(F)が Ok なら UserD.Txt をリセット}
    Timer1.Interval:=0;                              { タイマーを無効に設定する}
    Readln(F,UsrD[0]);                   { ファイルの第１行（データ番号 ）読込}
    Readln(F,UsrD[1]);                   { ファイルの第２行（X 軸機械座標）読込}
    Readln(F,UsrD[2]);                   { ファイルの第３行（Y 軸機械座標）読込}
    Readln(F,UsrD[3]);                   { ファイルの第４行（Z 軸機械座標）読込}
    Readln(F,UsrD[4]);                   { ファイルの第５行（A 軸機械座標）読込}
    Readln(F,UsrD[5]);                   { ファイルの第６行（IN0～3 ﾃﾞｰﾀ）読込}
    CloseFile(F);                                    { UserD.Txt をクローズ}
    Label1.caption:='データ番号= '+UsrD[0];          { データ番号を表示 }
    update;                                          { 情報をリフレッシュする }
    if UsrD[0]='1'then                               { データ番号が 1 の時の処理}
     begin
      Xpoint1:=StrToFloat(UsrD[1]);                  { X 軸座標を数値に変換記憶}
      AssignFile(F,Pas+'099.Txt');                   { 099.Txt ファイルを選択  }
      Rewrite(F);                                    { 099.Txt ファイル初期化  }
      Writeln(F,'(ユーザ作成プログラム１)');         { 第１行にタイトル書込み }
      Writeln(F,'G90G53G01X0F100');                  { 第２行に X 軸原点移動指令}
      Writeln(F,'G91G01G31X10F20');                  { 第３行にスキップ指令書込}
      Writeln(F,'G400R2');         { 番号２ユーザプログラムを処理させる指令書込}
      Writeln(F,'G200R99');        { 番号２で作成するプログラム２の実行指令書込}
      Writeln(F,'M02');                              { 最終行に M02 を書込み }
      CloseFile(F);                                  { 099.Txt ファイルクローズ}
      DeleteFile(Pas+'UserD.Txt');           { UserD.Txt を削除し 4W に制御を移す}
      update;                                        { 情報をリフレッシュする }
      Timer1.Interval:=50;                           { タイマーを 50ms に再設定 }
     end;
    if UsrD[0]='2'then                               { データ番号が 2 の時の処理}
     begin
      Xpoint2:=StrToFloat(UsrD[1]);                  { X 軸座標を数値に変換記憶}
      stra:=FloatToStr((Xpoint1+Xpoint2)/2);         { X 軸座標を平均し文字変換}
      AssignFile(F,Pas+'099.Txt');                   { 099.Txt ファイルを選択  }
      Rewrite(F);                                    { 099.Txt ファイル初期化  }
      Writeln(F,'(ユーザ作成プログラム２)');         { 第１行にタイトル書込み }
      Writeln(F,'G90G53G00X'+stra);                  { X 軸を平均計算位置に移動}
//    Writeln(F,'G200R2');                           { 他のプログラムに連結可能}
      Writeln(F,'M02');                              { 最終行に M02 を書込み }
      CloseFile(F);                                  { 099.Txt ファイルクローズ}
      DeleteFile(Pas+'UserD.Txt');           { UserD.Txt を削除し 4W に制御を移す}
      update;                                        { 情報をリフレッシュする }
      Close;                  { ユーザプログラム UserP.Exe を自動的に終了させる }
     end;
 end;

end.
```

６・４　遠隔制御運転プログラム例

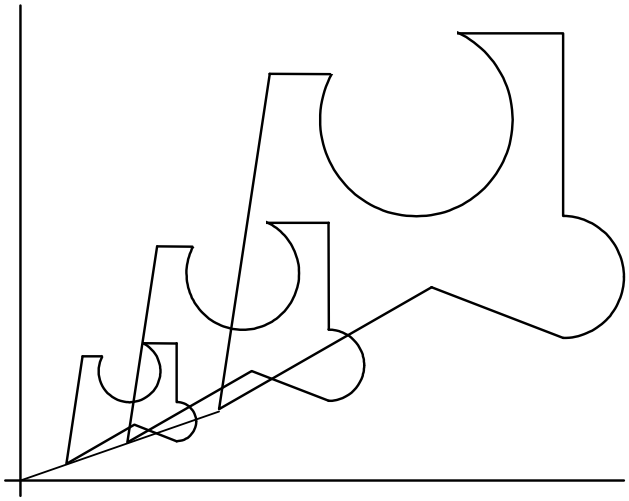
【プログラム例】（ファイル名 =DNC.ncm）

```
(TEST OF DNC).....(解り易いタイトル名を１行目に付ける)
 G91 G00 X100.....(早送りで X 軸 100mm 移動)
 G300.....(旧 DNC.ncm ファイルが削除され、次の遠隔制御を有効にします)
 M02.....(NC プログラム終了)
```

【ホスト側プログラム例】（Delphi を使用していますが他の言語でも可能）

```
Procedure Tform1. Button1Click(Sender: Tobject); .....{ボタン１をクリック}
   var F:TextFile;
  begin
   if FileExists('D:¥DNC.ncm')=False then .....{ファイルが無い時}
  begin
   AssignFile(F,'D:¥DNC.ncm'); .....{ファイルを選択}
   Rewrite(F); .....{ファイルを初期化}
   Writeln(F,'TEST OF DNC'); .....{１行目にタイトル書き込み}
   Writeln(F,'G91G00X100'); .....{２行目から加工プログラム}
   Writeln(F,'M02'); .....{最後に M02 をプログラム}
   CloseFile(F); .....{ファイルをクローズ}
  end;
end;
```

6･5　スケーリングプログラム例



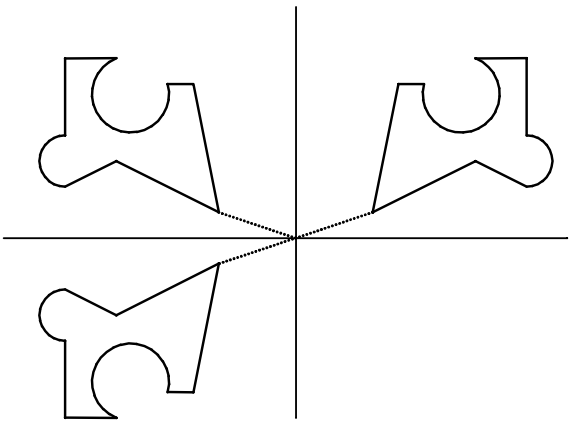
ソースプログラム

```
TEST SCALING Program
G92X0Y0
M98P1000
G51X0Y0P2
M98P1000
G50
G51X0Y0P0.5
M98P1000
G50
M30

O1000
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
M99
```

変換後プログラム

```
TEST SCALING Program
G92X0Y0
( Start SUB  P1000 L1 Nest1 )
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
( End SUB  P1000 )
( ** scaling  X0Y0  P=2.000 ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X60Y20
G01X80Y40F900
X40Y-20
G03Y40R20
G01Y60
X-40
G02X-40Y-20R-30
G01X-20
X-20Y-100
G00X-60Y-20
( End SUB  P1000 )
( *** scaling cancel *** )
( ** scaling  X0Y0  P=0.500 ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X15Y5
G01X20Y10F900
X10Y-5
G03Y10R5
G01Y15
X-10
G02X-10Y-5R-7.500
G01X-5
X-5Y-25
G00X-15Y-5
( End SUB  P1000 )
( *** scaling cancel *** )
M30
```

6･6　プログラマブルミラーイメージプログラム例



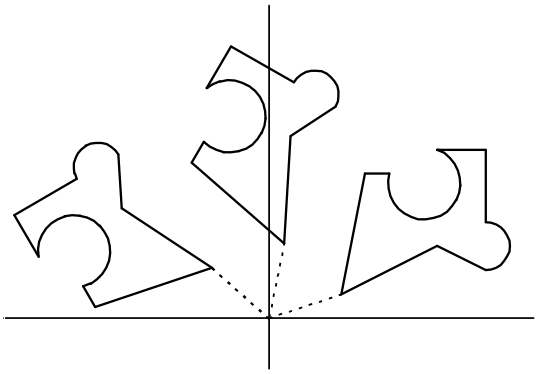
ソースプログラム

```
TEST MIRROR_IMAGE Program
G92X0Y0
M98P1000
G51.1X0
M98P1000
G50.1
G51.1X0Y0
M98P1000
G50.1
M30

O1000
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
M99
```

変換後プログラム

```
TEST MIRROR_IMAGE Program
G92X0Y0
( Start SUB  P1000 L1 Nest1 )
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
( End SUB  P1000 )
( ** mirror_image X0 ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X-30Y10
G01X-40Y20F900
X-20Y-10
G02Y20R10
G01Y30
X20
G03X20Y-10R-15
G01X10
X10Y-50
G00X30Y-10
( End SUB  P1000 )
( *** mirror_image cancel *** )
( ** mirror_image X0Y0 ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X-30Y-10
G01X-40Y-20F900
X-20Y10
G03Y-20R10
G01Y-30
X20
G02X20Y10R-15
G01X10
X10Y50
G00X30Y10
( End SUB  P1000 )
( *** mirror_image cancel *** )
M30
```

６･７　座標回転プログラム例



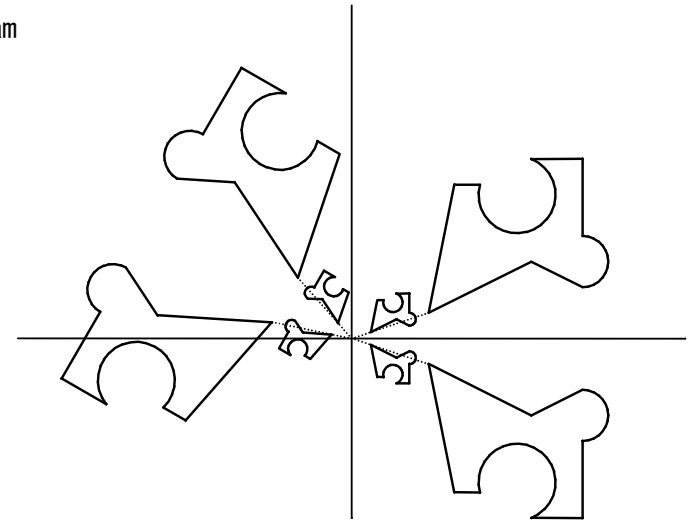
ソースプログラム

```
TEST COORDINATE ROTATE
G92X0Y0
M98P1000
G68X0Y0R60
M98P1000
G69
G68X0Y0R120
M98P1000
G69
M30

O1000
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
M99
```

変換後プログラム

```
TEST COORDINATE ROTATE Program
G92X0Y0
( Start SUB  P1000 L1 Nest1 )
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
( End SUB  P1000 )
( ** coordinate rotate  X0Y0  R=60.000deg ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X6.340Y30.981
G01X2.679Y44.641F900
X18.660Y12.320
G03X-17.320Y10R10
G01X-25.981Y15
X-10Y-17.320
G02X-1.340Y-22.321R-15
G01X-5Y-8.660
X38.302Y-33.660
G00X-6.340Y-30.981
( End SUB  P1000 )
( *** rotation cancel *** )
( ** coordinate rotate  X0Y0  R=120.000deg ** )
( Start SUB  P1000 L1 Nest1 )
G91G00X-23.660Y20.981
G01X-37.321Y24.641F900
X-1.340Y22.320
G03Y-17.320Y-10R10
G01X-25.981Y-15
X10Y-17.320
G02X18.660Y-12.321R-15
G01X5Y-8.660
X48.302Y16.340
G00X23.660Y-20.981
( End SUB  P1000 )
( *** rotation cancel *** )
M30
```

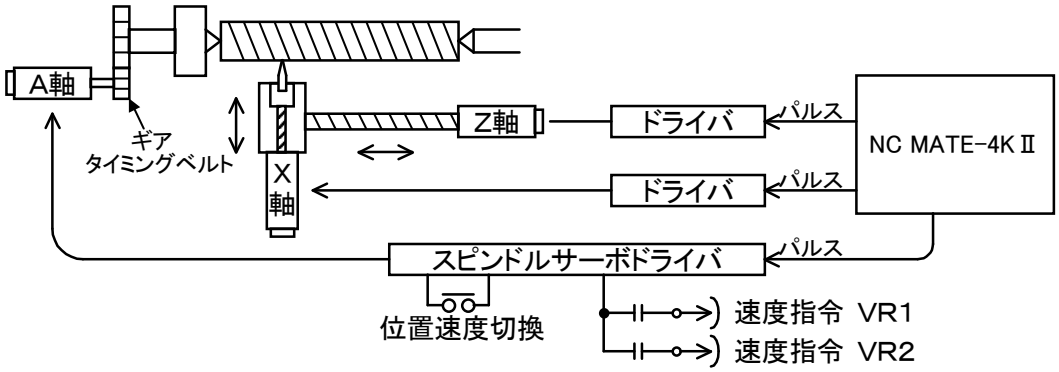
6・8　組み合わせプログラム例



```
ROTATE SCALING MIRROR Program
G92X0Y0
M98P101
G68X0Y0R150     ( ROTATE )
M98P101
G69
M30

O100            ( SCALING )
M98P1000
G51X0Y0P4
M98P1000
G50
M99

O101            ( MIRROR IMAGE )
M98P1000
G51.1Y0
M98P100
G50.1
M99

O1000           ( WORK SUB PROGRAM )
G91G00X30Y10
G01X40Y20F900
X20Y-10
G03Y20R10
G01Y30
X-20
G02X-20Y-10R-15
G01X-10
X-10Y-50
G00X-30Y-10
M99
```

6・9　同期制御例

同期制御でカム加工や割り出し，ネジ切り，リジットタップ加工など、主軸と同期して制御する必要がある場合には、主軸に位置制御と速度制御が切り換え可能なモータを採用してください。

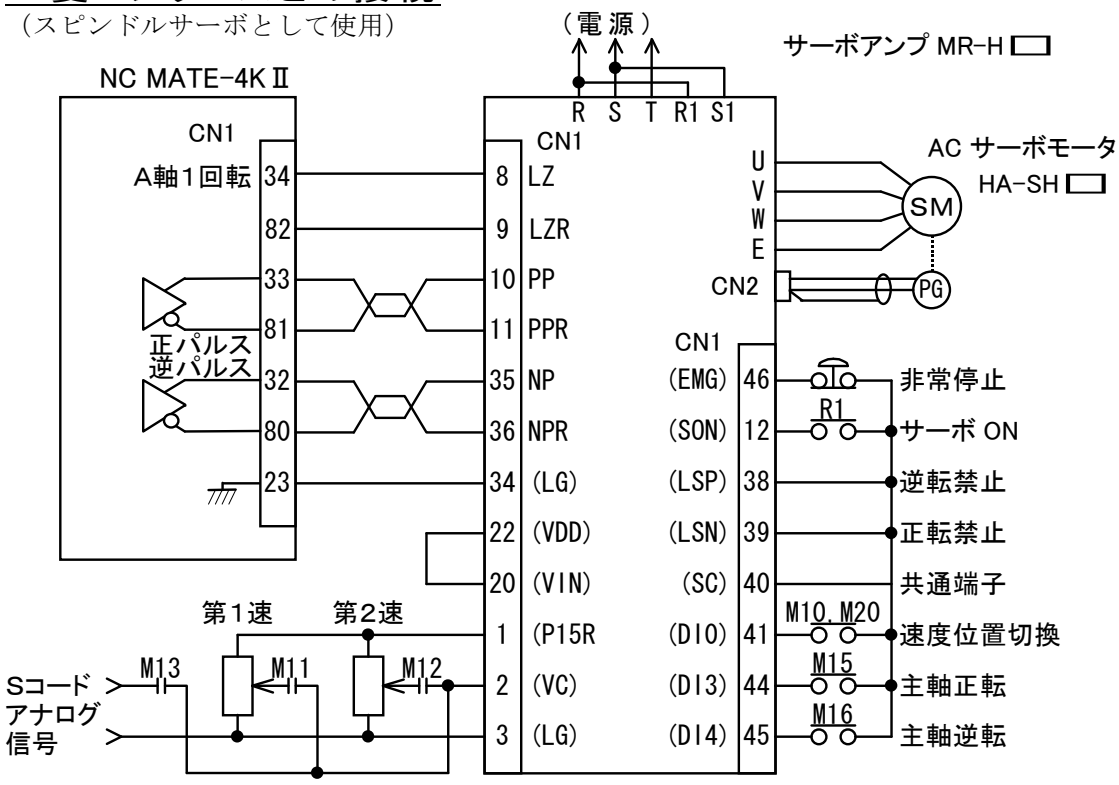


※A 軸をスピンドルサーボ方式にしますと、主軸エンコーダやインターフェイスが不要となり、コスト的にも性能的にも有利となります。

※X 軸の伝導系は半径指定（１パルス当たり 0.001mm 移動）です。

三菱Hシリーズとの接続

（スピンドルサーボとして使用）



6・9・1　ネジ切り加工プログラム例

```
M20 ･･･････････････････････ Ａ軸位置制御切り換え（A1.0 で主軸１回転）
G28.4 ･･･････････････････････ Ａ軸自動原点復帰（常時減速ドグ検出方式）
G91 G01 X-1.5 A6.0 F309 ･････ 径方向へ 1.5ｍｍ 切り込み（Ａ軸回転速度＝
300rpm）
G01 Z-20.0 A10.0 F671 ･･･････ P＝2ｍｍ，ネジ長 20ｍｍ 加工
G01 Z-2.0 A1.0 X1.5 F807 ････ ネジの切り上げ
G01 Z22.0 A11.0 F671 ･･･････ Ｚ軸スタート点へ
G01 X-2.0 A6.0 F316
G01 Z-20.0 A10.0 F671
G01 Z-2.0 A1.0 X2.0 F900
G01 Z22.0 A11.0 F671
M10 ･･･････････････････････ Ａ軸速度制御切り換え（毎分送り加工）
M12 ･･･････････････････････ 回転数（第２速）を選択
M15 ･･･････････････････････ Ａ軸正転
 :
 :･･･････････････････････ 外型加工など毎分送り加工
 :
M02
```

※ Ａ軸(主軸)の回転速度は一定速が良いため、下記計算で F 速度を指定して下さい。

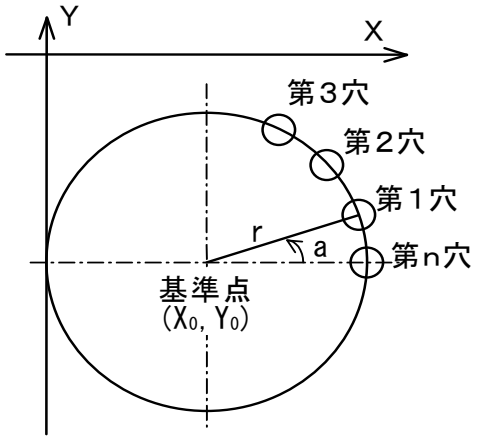
ネジ切り時の　$F=(\sqrt{X^2+Z^2+A^2}\div A)\times$ Ａ軸回転速度（一定速が良い）

Ｘ軸移動量 1.5ｍｍ，Ｚ軸 -2ｍｍ ，Ａ軸回転量１回 ，Ａ軸回転速度 300rpm の場合

送り速度指令値　$F=(\sqrt{(1.5)^2+(-2)^2+(1)^2}\div 1)\times 300=807$

6・10　自動プログラム開発例

6・10・1　ボルトホールサークルの自動プログラム(カスタムマクロ)開発例



	作成先とファイル名	BoltHole.Txt
X0	基準点のＸ座標値	100
Y0	基準点のＹ座標値	-200
r	半　径	100
a	始　角	10
n	穴個数	36

ファイル作成

このソフトは、簡単なデータ入力で加工プログラムを作成できる自動プロ開発例です。
使用方法は、この自動プロを立ち上げて必要なデータを入力し「ファイル作成」押ボタンを押すと、指定された作成先に加工プログラム BoltHole.Txt が記録されます。
このファイルを NC MATE-4KⅡで読み込む事により自動運転ができます。

ボルトホールサークルのソフト（Delphi を使用していますがエクセルのマクロでも作成可能）

```
procedure TForm1.Button1Click(Sender: TObject);{ファイル作成}
  var F:TextFile; str,xdata,ydata:string;
  v1,v2,v3,v4,v6,v100,v101,v102,v103,v104:double;
 begin
   str:=Edit1.Text; AssignFile(F,str);     {ファイルを選択}
   Rewrite(F);                             {ファイルを初期化}
   Writeln(F,'('+str+')');                 {１行目にタイトル書込}
   v1:=StrToFloat(Edit2.Text);             {基準点のＸ座標を読込}
   v2:=StrToFloat(Edit3.Text);             {基準点のＹ座標を読込}
   v3:=StrToFloat(Edit4.Text);             {半径を読込}
   v4:=StrToFloat(Edit5.Text);             {始角を読込}
   v6:=StrToFloat(Edit6.Text);             {穴個数を読込}
   v100:=0; v101:=abs(v6);                 {初期値設定と絶対化}
   repeat
    v102:=v100*360/v6; v102:=v4+v102;          {穴位置角度計算}
    v103:=v3*cos(v102*pi/180); v103:=v1+v103;      {Ｘ座標計算}
    v104:=v3*sin(v102*pi/180); v104:=v2+v104;      {Ｙ座標計算}
    xdata:=FloatToStrF(v103,ffFixed,11,3);    {Ｘデータ文字変換}
    ydata:=FloatToStrF(v104,ffFixed,11,3);    {Ｙデータ文字変換}
    Writeln(F,'G90G00'+'X'+xdata+'Y'+ydata);   {ファイルに書込}
    Writeln(F,'M10');                          {M10 をファイルに書込}
    v100:=v100+1;                           {穴個数の積算}
   until v100>=v101;                        {穴個数の判断}
   Writeln(F,'M02');                        {最後に M02 をプログラム}
   CloseFile(F);                            {ファイルをクローズ}
 end;
end.
```

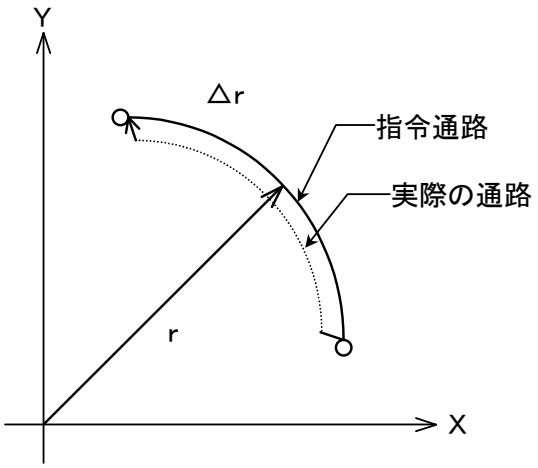
6･10･2　ボルトホールサークルの自動プログラム(エクセルマクロ)開発例

このソフトは、簡単なデータ入力で加工プログラムを作成できる、エクセルのマクロを使用した自動プログラム開発例です。
使用方法は、エクセルのマクロでこの自動プログラムを立ち上げて、必要なデータを入力し「ファイル作成」押ボタンを押すと、指定された作成先に、加工プログラム BoltHole.Txt が記録されます。このファイルを NC MATE-4KⅡで読み込む事により自動運転ができます。

```
Private Sub Command ファイル作成_Click()

Dim FileName As String, xdata As String, ydata As String
Dim v1 As Double, v2 As Double, v3 As Double, v4 As Double
Dim v6 As Double, v100 As Double, v101 As Double, v102 As Double
Dim v103 As Double, v104 As Double
Const pi As Double = 3.141592654

FileName = TextBox1.Text                    '作成先とファイル名
If FileName = "" Then Exit Sub              '空白のとき
v1 = CDbl(TextBox2.Text)                    '基準点の X 座標
v2 = CDbl(TextBox3.Text)                    '基準点の Y 座標
v3 = CDbl(TextBox4.Text)                    '半径
v4 = CDbl(TextBox5.Text)                    '始角
v6 = CDbl(TextBox6.Text)                    '穴個数
v100 = 0                                    '初期化
v101 = Abs(v6)                              '絶対値

Open FileName For Output As #1              'ファイルを開く
Print #1, FileName                          '1 行目にタイトル書込
Do
    v102 = v100 * 360 / v6
    v102 = v4 + v102                        '穴位置角度計算
    v103 = v3 * Cos(v102 * pi / 180)
    v103 = v1 + v103                        'X 座標計算
    v104 = v3 * Sin(v102 * pi / 180)
    v104 = v2 + v104                        'Y 座標計算
    xdata = Format(v103, "###0.000")
    ydata = Format(v104, "###0.000")
    Print #1, "G90G00X" & xdata & "Y" & ydata   'ファイルに書込
    Print #1, "M10"                         'M10 をファイルに書込
    v100 = v100 + 1                         '穴個数の積算
Loop Until v100 >= v101                     '穴個数の判断
Print #1, "M02"                             '最後に M02 をファイルに書込
Close #1                                    'ファイルを閉じる
End Sub


Private Sub UserForm_Click()

End Sub
```

6・11　円弧切削の場合の半径方向誤差について

サーボモータを使用する時には、指令と出力軸の間には位置決め系による遅れが生じます。
直線補間の場合は、指令された線分上を進むので誤差は生じませんが、円弧補間の場合、
特に高速の円弧切削を行う場合には、加速度の関係で半径方向に誤差が生じます。
そして、この誤差は次のようにして求めることができます。



△ｒ：半径誤差の最大値（mm）
ｖ：送り速度（mm/sec）
ｒ：円弧半径（mm）
T1：切削時の exponential 加減速度の時定数（sec）
T2：位置決め系の時定数（sec）
（ポジションループゲインの逆数）

$$\triangle r = \frac{1}{2}(T1^2 + T2^2) \cdot \frac{v^2}{r} \quad \cdots\cdots\cdots (1)$$

実際の加工を行う場合には、工作物の加工半径 ｒ（mm）と許容誤差△ ｒ（mm）が与えられるので、（1）式において許容限界速度 ｖ（mm/sec）が決まります。
なお、本装置で設定されている切削時の加減速時定数は機械によって異なるので、機械メーカー発行の説明書を参照してください。

【計算例】
送り速度 200mm/min. で、半径 10mm の円弧切削を行う場合の誤差

T1 ＝加減速時定数（スムージング時定数）は 10.24mS ～ 1000mS（計算例 25mS）
T2 ＝サーボ時定数はループゲイン KP ＝ 30 の時 33mS（T2 ＝ 1/KP）
△ ｒ ＝ $\{(0.025)^2 + (0.033)^2\} \cdot (200/60)^2 \times \fallingdotseq 0.00095$mm

【備考】
NC MATE-4K Ⅱのパルス分配による誤差は、上記サーボモータに起因する誤差に比べ小さいので、計算式に含まれません。

第７章　困ったときには…

7・1　トラブル発生時のチェック要領

NC MATE-4K Ⅱを使用したシステムでハードに起因するトラブルが発生した場合、下記の要領で適切な処置を行ってください。

【チェック 1】

トラブルが機械系より発生しているか、電気系から発生しているかを明確にする。

【チェック 2】

配線が正確に接続されているかどうか？時々断線していないかチェックする。

【チェック 3】

通常は正確に動作して、時々異常な動作またはダウンする症状でトラブルが同期的に発生するか、不規則に発生するかをチェックする。

正確に位置決めがされない時がある。

①モータ軸に合印マークを付け、プログラムまたは手動送りにてモータを回転させ、カップリングやタイミングベルトにずれがないかチェックしてください。
特に急な加減速を伴う減速機付サーボモータの場合、減速機内の軸とギア間にずれが発生する可能性があります。

② NC MATE-4K Ⅱ基板，とサーボドライバーの指令パルス体系が一致しているかチェックしてください。

移動限でない時にオーバートラベルが発生する。

①入力状態画面にして、各リミットスイッチや押ボタンからの信号を電線やリミットスイッチを動かしながらチェックしてください。

② DC 5V（±0.2V），DC 24V（±1V）の電圧が正常か？ リップルや寸断がないかどうかテスターやオシロスコープでチェックしてください。

リレーやソレノイドバルブ，モータや他の機器の ON，OFF に同期して、時々異常が発生する。

①この場合はほとんどノイズによるトラブルですから、リレーやソレノイド，モータにノイズの発生を防止するサージキラーを必ず取り付けてください。

②プラズマ切断器や溶接器など、非常に強力なノイズ発生源がある場合、近くを通る配線は全てシールド処理をして接地工事をするとともに、別電源を使用してください。

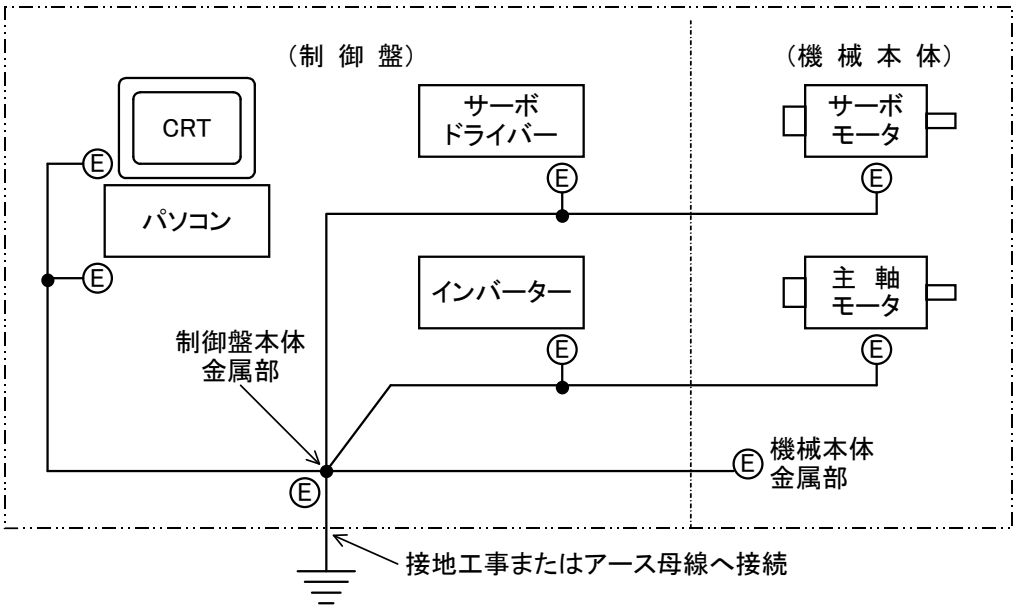
③ノイズの有無をチェックするには、トランジスターラジオを制御盤内に入れ、無選局の状態にしますとノイズ発生時にクリック音が出ますので、ノイズ発生の有無がチェックできます。

最近、時々異常な動作が不規則におこるようになってきた。

この原因は、ほとんどコネクター部の接触不良です。NC MATE-4K Ⅱの接続部を洗浄剤などで洗浄してください。

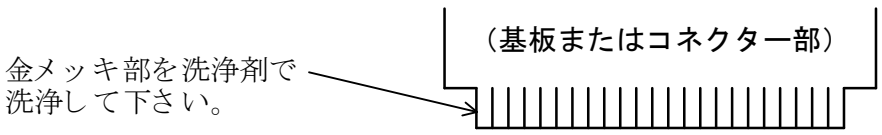
アースが不完全なため、漏電流により予期せぬ所へ電流が流れ、誤動作や基板内の IC が破損した。

上記トラブルの発生を防ぐためには、下図のように各機器のアース端子は必ず結線し、制御盤で１ヵ所にまとめ、接地してください。



NC MATE-4K Ⅱ基板の接続部が、微振動やヒューム，ガスなどの侵入により接触不良になり、時々誤動作やシステムがダウンした。

この場合、接続ケーブルのコネクタ部を動作中に動かし、接触不良を再現させることによりチェックすることが可能です。
上記トラブルの発生を防ぐためには、コネクタ部を電子機器用の洗浄剤で完全に洗浄して結合し、結合箇所からヒュームやガスが侵入しないようにします。更に振動により結合部が相対的に動かないように固定ネジなどは確実にロックしてください。



ハンダ付け不良や圧着端子の接続不良などにより結線が完全でないため、誤動作やアラームが発生する。

上記トラブルの発生を防ぐためには、ハンダ付けの場合、両方の接合部を別々にハンダメッキし、フラックスを十分に付けてハンダ付けしてください。
圧着端子の場合は、電線と端子を引っ張り、完全に圧着されているかのチェックを十分にしてください。
また、配線作業中にハンダや配線クズが基板上に付着し、トラブルの原因になることがありますので、注意してください。

IC や電子部品（OSC，抵抗器，ディップスイッチ）の初期不良によるトラブル発生した。

このトラブルは主に１～６カ月間で発生致します。基板を弊社まで送付していただきましたら、無償にて新品と交換させていただきます。

サーボモータの感度調整が不完全なため、円弧切削や加減速時に加工精度が悪くなる。

〔サーボモータの主な感度の調整要領〕

ａ）位置ループゲイン .....（通常 25 ～ 100 で補間軸のゲインはあわせる必要がある）
ｂ）速度ループゲイン .....（安定な範囲で高いゲインに設定する必要がある）
ｃ）速度積分時定数 .....（安定な範囲でなるべく小さく設定する必要がある）
ｄ）位置指令加減速時定数 .....（クッションの時定数で 10 ～ 30mS に設定する）
上記の項目を中心に、振動やハンチング，加減速時のショックなどが発生しない範囲で、なるべく高感度に調整してください。

サーボモータ１回転信号の出力がオープンコレクター方式の時、供給電流（定格 15mA）が過大になる電圧（5V 以上）を使用した。

この場合、NC MATE-4K Ⅱ基板内の抵抗（330Ω）が不良になりますので、供給電流が過大になる電圧の使用しないでください。

原点復帰が不安定なため初回に誤作が発生する。

これは、原点減速ドグの長さが不足したり、減速リミットの取り付けが不適当などにより原点位置がモータ１回転の移動量だけずれることにより発生します。
これらのトラブルを防ぐためには、原点マークを機械の適所に取り付けて、作業開始のとき大幅にずれていないことが確認できるようにしてください。

サーボモータの軸径より太くて長いボールネジを直結された。そのためサーボのゲインを基準値まで上げることができず、結果的に加工精度や加工面が悪い。

上記トラブルの発生を防ぐためには、タイミングベルトで 1/2 ～ 1/3 に減速し、負荷慣性をできるだけ小さくしてください。(P. 131 「サーボモータ概略選定の手引」参照)

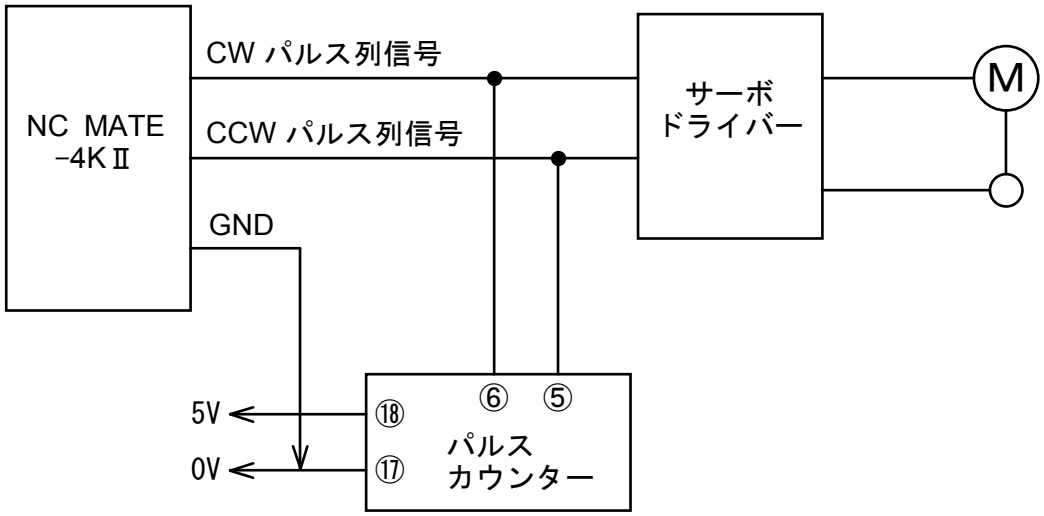
サーボモータのフィードバック線が時々断線したり、接触不良などにより CRT 画面の位置表示と実際のサーボモータの移動量が異なるトラブルが発生した。

上記トラブルは時々しか発生しないため原因調査が困難ですが、入力パルス数カウンタが内蔵されているサーボモータがありますので、この値と CRT の位置表示とサーボモータの回転角をチェックすることで、どこにトラブルの原因があるか判断可能です。

〔チェック方法〕CRT およびカウンターの表示値を「0」にしてシステムを運転する。

① CRT の位置とサーボ内蔵カウンターが同じで、モータ回転角が異なる場合
　→サーボモータ，サーボドライバー，フィードバック配線の不良
② CRT 位置表示とサーボ内蔵カウンターが異なる場合
　→ NC MATE-4K Ⅱ，パルス信号回路などの不良
※ただし、バックラッシュ補正やピッチ誤差補正が作動した場合は、その分だけ CRT 表示とカウンタ値は異なります。

サーボモータにカウンタが内蔵されていない場合、パルスカウンタを下図のように接続することで、上記のチェックが可能です。



手動パルスハンドル用電源（CN1 の 44 番、92 番）をショートさせたためシステムがダウンした。

基板より 5V を供給しているため、この配線をショートさせますと基板が故障しますので注意してください。
このトラブルを避けるには、別電源を使用するか、または NCMATE-4K Ⅱ と手動パルスハンドル間に 0.1A 程度のヒューズを入れてください。

7・2　工具磨耗や破損時の自動運転再開方法

金型など大容量ステップの加工中に、工具磨耗や破損により自動運転が続行できない場合や、工具を交換して最初から自動運転をやり直すには、多くの時間が必要な場合は下記の方法で自動運転を再開してください。

①工具磨耗や破損トラブルが発生した直後であれば、自動運転を休止させ、各軸の座標をメモするか、または相対座標を 0 にした後、工具交換位置に手動モードで移動させます。工具を交換した後、運転を休止させた位置に戻すと自動運転が再開できます。

②工具破損トラブルが発生した数ブロック前から加工をやり直す必要がある場合は、現在のステップ数から推測して自動運転を再開するステップを計算します。

③工具交換位置に移動させて工具を交換し、プログラムのスタート点に移動させます。

④軌跡シミュレーションやマシンロックでトラブル発生以前のステップの開始座標をメモします。（再開ステップに近くなったらシングルブロックを ON にすると良い）

⑤自動運転を再開するステップの開始座標まで工具位置を手動や MDI で移動させます。

⑥マシンロックを ON にして、プログラムのスタート点に表示だけを移動させます。

⑦プログラムを再度読み込み、マシンロックを ON にして、自動運転をスタートさせ再開します。ステップに近くなったらシングルブロックを ON にし、再開ステップで止めます。

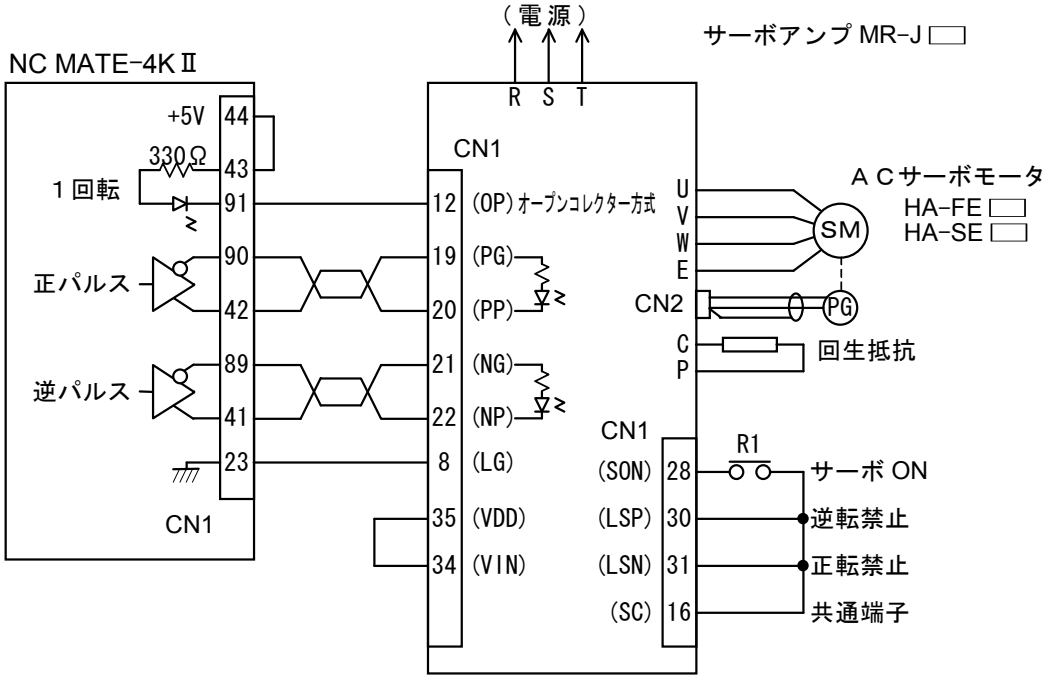
⑧マシンロックとシングルブロックを OFF にして、自動運転をスタートさせます。

マシンロックは移動を伴いませんので短い時間で再開ステップに到達しますが、M コードなど補助機能は実行されますので注意してください。

第 8 章　参考資料

8・1　各社モータとの接続例

8・1・1　三菱 J シリーズとの接続



【備考】

・三菱 J シリーズの最大入力パルス周波数は 200KPPS 以下です。
（1 パルス 0.001mm の時 早送り速度 Max.12m/min.）

・三菱 J シリーズ参考パラメータ値

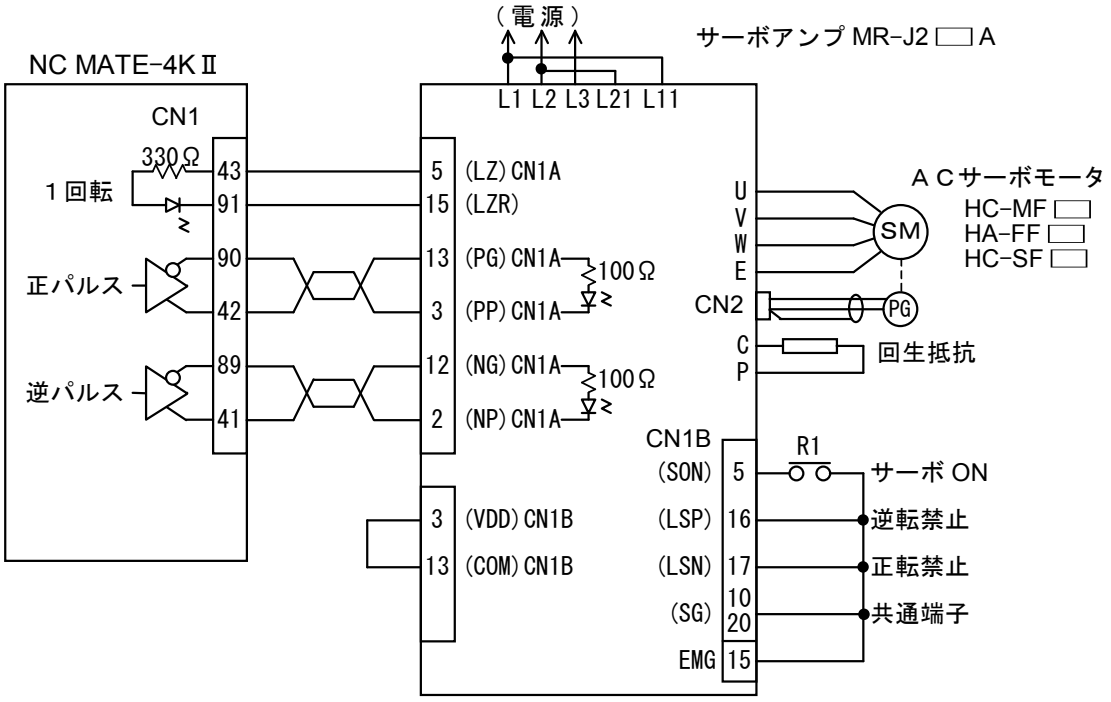
※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

$$\frac{Pr2}{Pr3}=\frac{\text{設定単位}\times\text{検出器のフィードバックパルス}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times4000}{10}=\frac{4}{10}=\frac{2}{5}$$

ただし $1/50 < \frac{Pr2}{Pr3} < 20$ の範囲で設定

内容		設定値
指令パルス列形態	正パルス，逆パルス方式	Pr7 ＝ 0000
位置指令加減速時定数	10 ～ 30mS	Pr10 ＝ 1 ～ 3
位置ループゲイン	安定な範囲で大きく	Pr 5＝ 25 ～ 100
速度ループゲイン	安定な範囲で大きく	Pr12 ＝ 100
速度積分補償	安定な範囲で小さく	Pr13 ＝ 10
モニター切り換え	指令パルス累積値	Pr20 ＝ 5
フィードフォワード制御	安定な範囲で設定	Pr8 ＝ 0 ～ 25

8・1・2　三菱 J2 シリーズとの接続



【備考】

・三菱 J2 シリーズの最大入力パルス周波数は 400KPPS 以下です。
（1 パルス 0.001mm の時 早送り速度 Max.24m/min.）

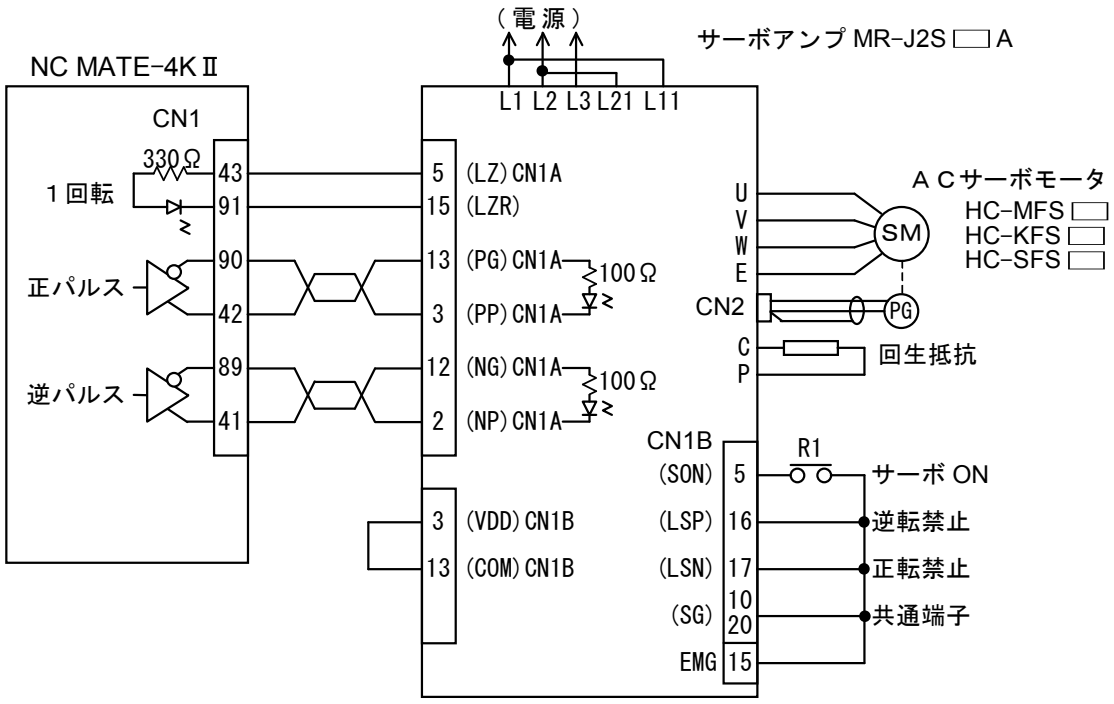
・三菱 J2 シリーズ参考パラメータ値
※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

$$\frac{CMX}{CDV}=\frac{\text{設定単位(mm)}\times\text{検出器のフィードバック(パルス)}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times 8192}{10}=\frac{8.192}{10}=\frac{4096}{5000}=\frac{512}{625}$$

ただし $1/50 < \frac{CMX}{CDV} < 50$ の範囲で設定

内容		設定値
指令パルス列形態	正パルス，逆パルス方式	OP3 ＝ 0010
位置指令加減速時定数	10 ～ 30mS	PST=10 ～ 30
位置ループゲイン	安定な範囲で大きく	PG1=25 ～ 100
速度ループゲイン	安定な範囲で大きく	VG1=216, VG2=714
速度積分補償	安定な範囲で小さく	VIC=10
モニター切り換え	指令パルス累積値	DMD=0003
オートチューニング	補間モード	ATU=0003

8・1・3　三菱 J2S シリーズとの接続



【備考】

・三菱 J 2 S シリーズの最大入力パルス周波数は 500KPPS 以下です。
（1 パルス 0.001mm の時 早送り速度 Max.30m/min.）

・三菱 J 2 S シリーズ参考パラメータ値

※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

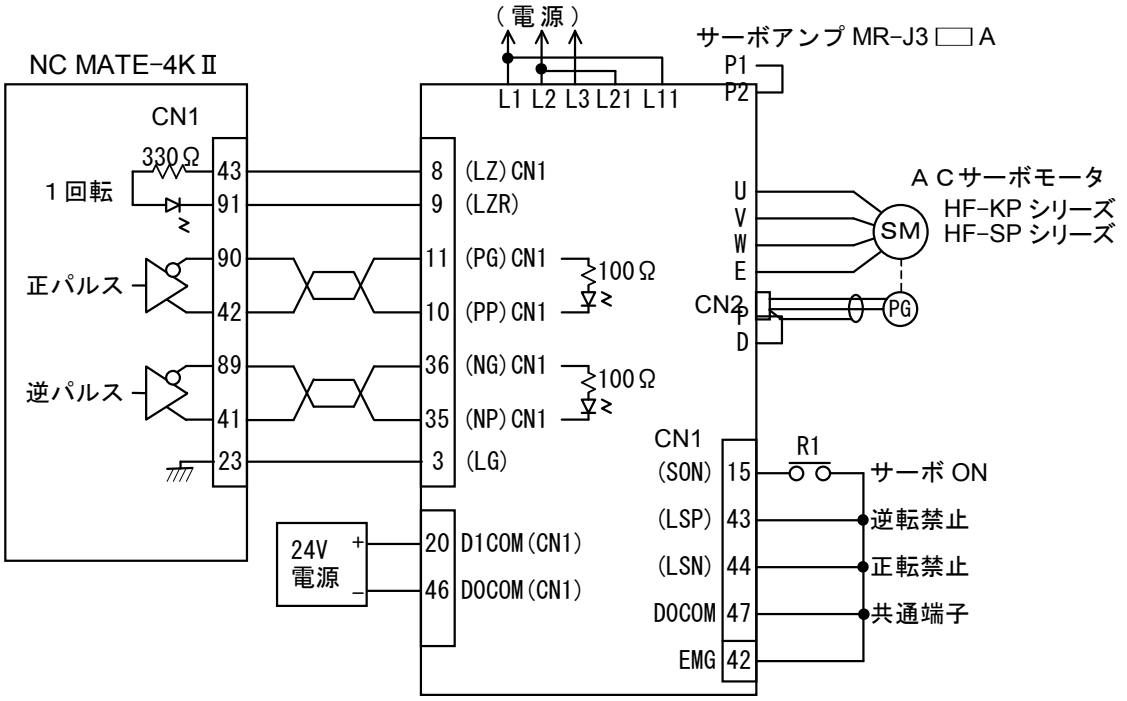
$$\frac{CMX}{CDV}=\frac{\text{設定単位(mm)}\times\text{検出器のフィードバック(ﾊﾟﾙｽ)}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times131072}{10}=\frac{131.072}{10}$$

$$=\frac{65536}{5000}=\frac{8192}{625}$$

ただし 1/50 ＜ $\frac{CMX}{CDV}$ ＜ 500 の範囲で設定

内容		設定値
指令パルス列形態	正パルス，逆パルス方式	OP3 = 0010
位置指令加減速時定数	10 ～ 30mS	PST=10 ～ 30（補間軸は同値）
位置ループゲイン	安定な範囲で大きく	PG1=25 ～ 100（補間軸は同値）
速度ループゲイン	安定な範囲で大きく	VG1=177, VG2=817
速度積分補償	安定な範囲で小さく	VIC=10
モニター切り換え	指令パルス累積値	DMD=0003
オートチューニング	補間モード	ATU=0005

8・1・4　三菱Ｊ３シリーズとの接続



【備考】

三菱 Ｊ ３シリーズの最大入力パルス周波数は 1000KPPS 以下です。

（1 パルス 0.001mm の時 早送り速度 Max.47m/min.）

三菱 Ｊ ３シリーズ参考パラメータ値

※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

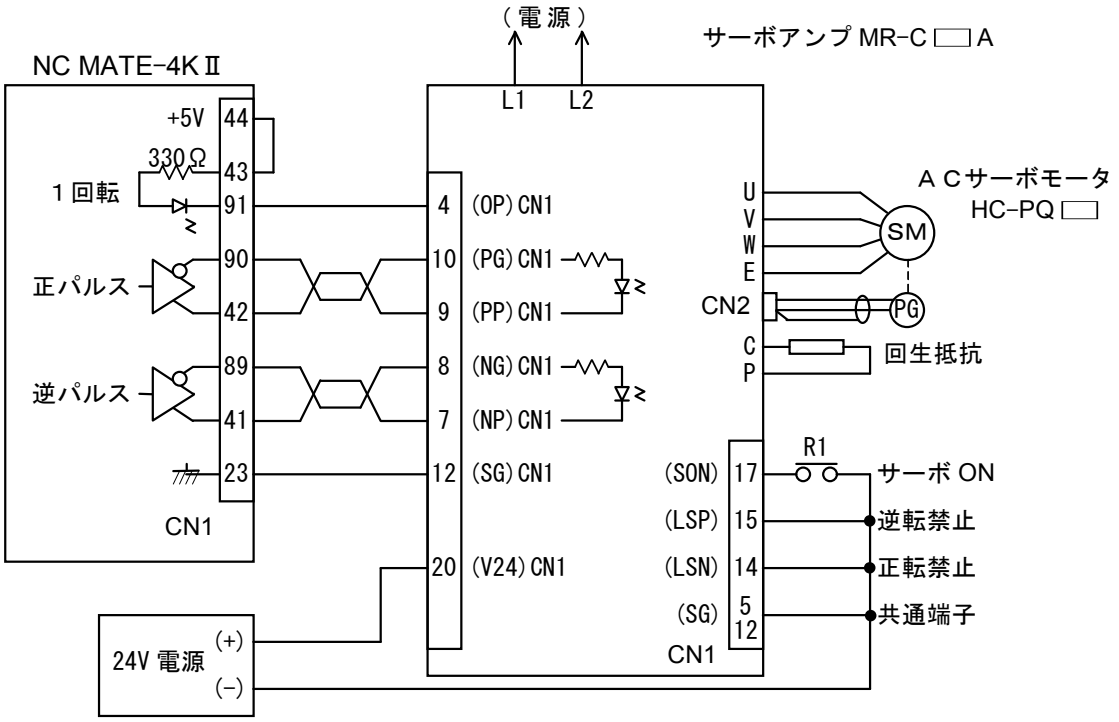
$$\frac{CMX}{CDV}=\frac{\text{設定単位(mm)}\times\text{検出器のフィードバック(ﾊﾟﾙｽ)}}{\text{モータ１回転当たりの移動量mm}}=\frac{0.001\times262144}{10}$$

$$=\frac{262144}{10000}=\frac{16384}{625}$$

ただし 1/10 ＜ $\frac{CMX}{CDV}$ ＜ 2000 の範囲で設定

内容		設定値
指令パルス列形態	正パルス，逆パルス方式	PLSS=0010
位置指令加減速時定数	10 ～ 30mS	PST=10 ～ 30（補間軸は同値）
位置ループゲイン	安定な範囲で大きく	PG1=24 ～ 100（補間軸は同値）
速度ループゲイン	安定な範囲で大きく	VG2=823
速度積分補償	安定な範囲で小さく	VIC=10
モニター切り換え	指令パルス累積値	DMD=0003
オートチューニング	補間モード	ATU=0000

8・1・5　三菱 C シリーズとの接続



【備考】

三菱 C シリーズの最大入力パルス周波数は 200KPPS 以下です。

（1 パルス 0.001mm の時 早送り速度 Max.12m/min.）

三菱 C シリーズ参考パラメータ値

※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

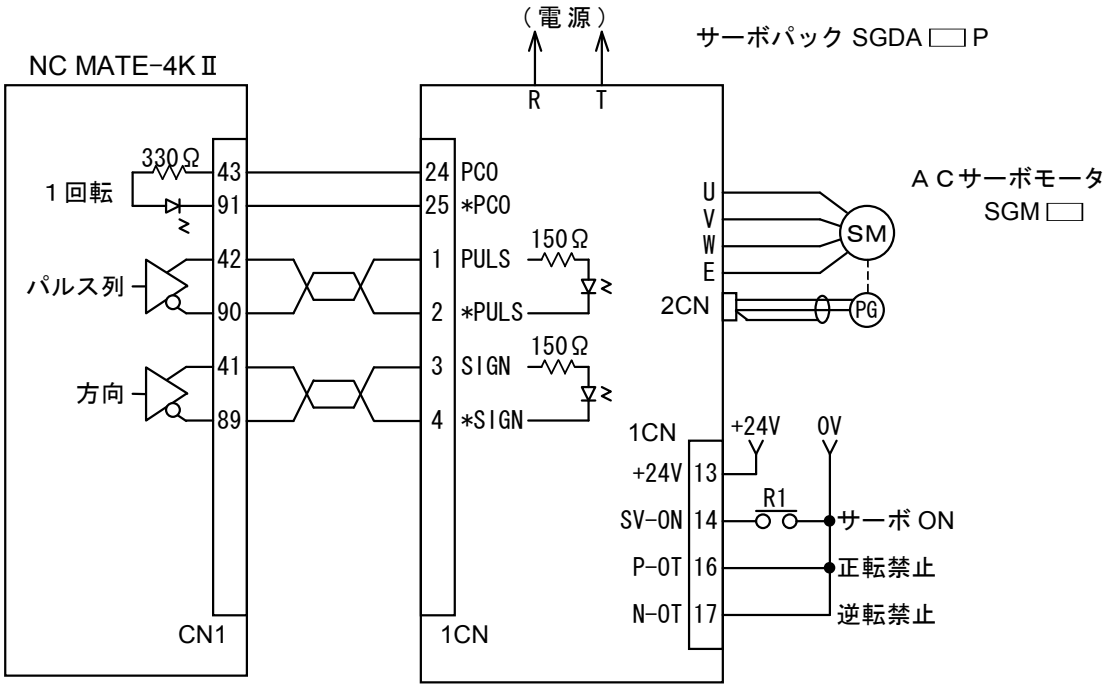
$$\frac{CMX}{CDV}=\frac{\text{設定単位(mm)}\times\text{検出器のフィードバック(パルス)}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times4000}{10}=\frac{4}{10}$$

ただし $1/50 < \frac{CMX}{CDV} < 50$ の範囲で設定

内容		設定値
指令パルス列形態	正パルス，方向方式	OP1=010
位置指令加減速時定数	10 ～ 30mS	RST=1 ～ 30
オートチューニング	オートチューニングしない	ATU=202

> ⚠ 直線補間や円弧補間などを使用する場合、補間軸の位置ループゲイン加減速時定数は合わせる必要がありますので、オートチューニングは使用できません。
> 位置ループ関係のパラメータは 25 番～ 30 番にありますが通常は設定できませんので、パラメータ 12 番を 00d にすると設定可能となります。
> 詳細は三菱サーボセンターまで。

8・1・6　安川 Σ シリーズとの接続



【備考】

・安川 Σ シリーズへの最大入力パルス周波数は 450KPPS 以下です。
（1 パルス 0.001mm の時 早送り速度 Max.27m/min.）

・安川 Σ シリーズ参考パラメータ値

※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

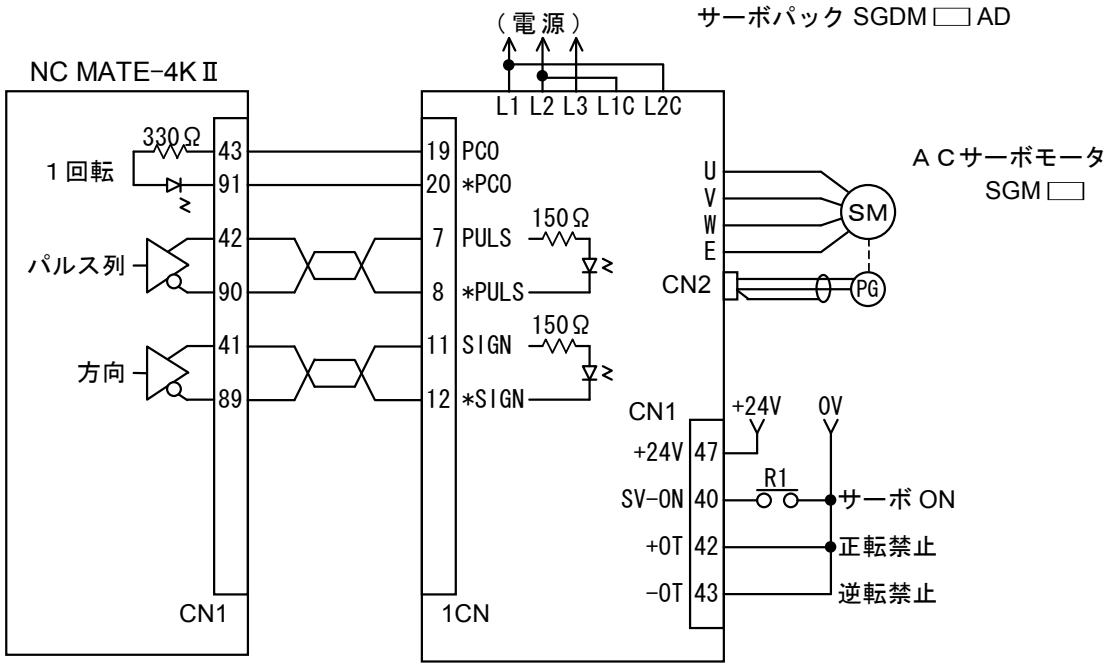
$$\frac{\text{Cn-24}}{\text{Cn-25}}=\frac{\text{設定単位}\times\text{検出器のフィードバックパルス}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times 8192}{10}=\frac{8.192}{10}=\frac{512}{625}$$

ただし $0.01 \leqq \frac{\text{Cn-24}}{\text{Cn-25}} \leqq 100$

内容		設定値
指令パルス列形態	パルス列、方向	Cn-02,3,4,5=0,0,0
スムージング時定数	10 ～ 30mS	Cn-26=100 ～ 300
位置ループゲイン	安定な範囲で大きく	Cn-1A=25 ～ 100
速度ループゲイン	安定な範囲で大きく	Cn-04=80
速度ループ積分時定数	安定な範囲で小さく	Cn-05=10

8・1・7　安川 Σ Ⅱシリーズとの接続

（



【備考】

・安川 Σ Ⅱシリーズへの最大入力パルス周波数は 450KPPS 以下です。
（1 パルス 0.001mm の時 早送り速度 Max.27m/min.）

・安川 Σ Ⅱシリーズ参考パラメータ値

※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

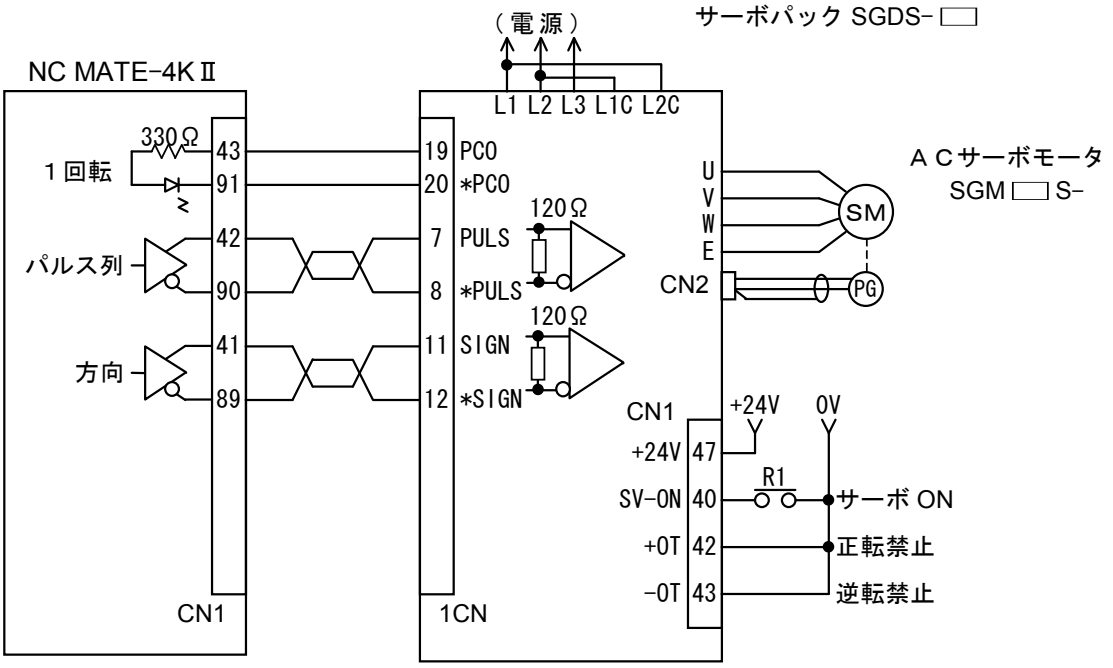
$$\frac{Pn202}{Pn203}=\frac{\text{設定単位}\times\text{検出器のフィードバックパルス}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times 8192}{10}=\frac{8.192}{10}=\frac{512}{625}$$

ただし $0.01 \leqq \frac{Pn202}{Pn203} \leqq 100$

内容		設定値
指令パルス列形態	正パルス，方向方式	Pn200=0000
スムージング時定数	10 ～ 30mS	Pn204=1000 ～ 3000 （補間軸は同値）
位置ループゲイン	安定な範囲で大きく	Pn102 ＝ 25 ～ 100 （補間軸は同値）
速度ループゲイン	安定な範囲で大きく	Pn100=40 ～ 80
速度ループ積分時定数	安定な範囲で小さく	Pn101=1000 ～ 2000

8・1・8　安川 Σ Ⅲシリーズとの接続

（



【備考】

安川 Σ Ⅲシリーズへの最大入力パルス周波数は 800KPPS 以下です。

（1 パルス 0.001mm の時 早送り速度 Max.47m/min.）

・安川 Σ Ⅲシリーズ参考パラメータ値

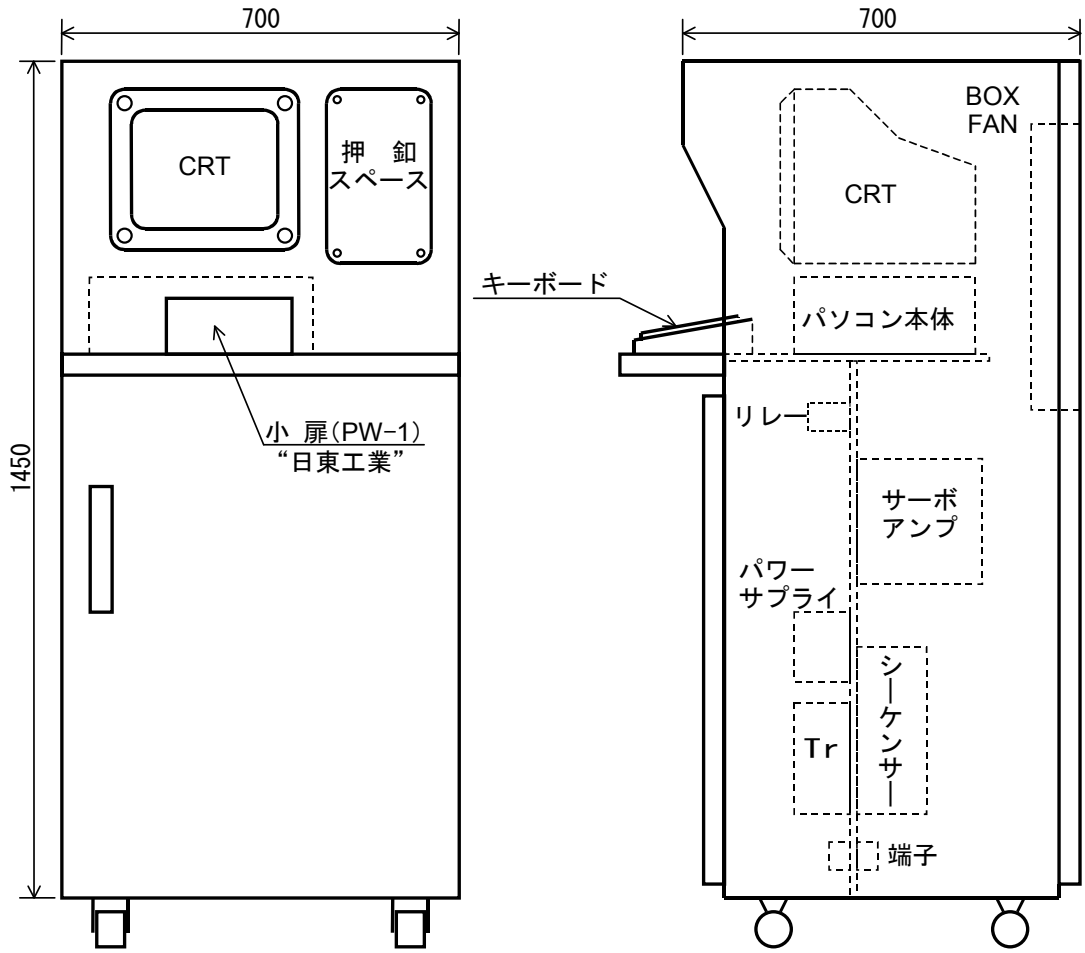
※ピッチ＝ 10mm のボールネジに直結の時の電子ギア設定値

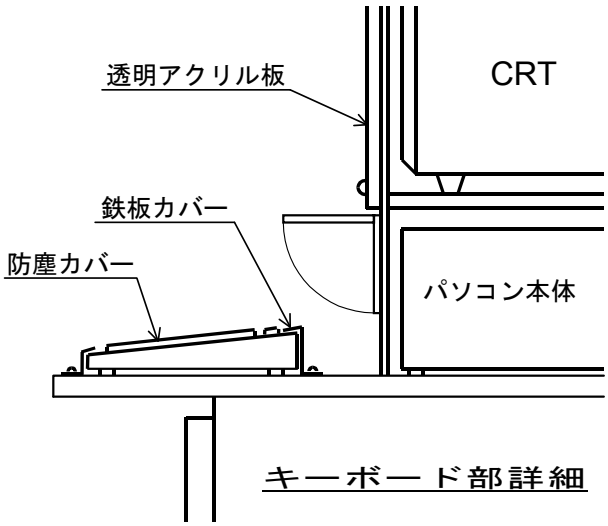
$$\frac{\text{Pn20E}}{\text{Pn210}}=\frac{\text{設定単位}\times\text{検出器のフィードバックパルス}}{\text{モータ 1 回転当たりの移動量mm}}=\frac{0.001\times131072}{10}=\frac{131.072}{10}=\frac{8192}{625}$$

ただし $0.001 \leqq \frac{\text{Pn20E}}{\text{Pn210}} \leqq 1000$

内容		設定値
指令パルス列形態	正パルス，方向方式	Pn200= ロロロ 0
スムージング時定数	10 ～ 30mS	Pn216=1000 ～ 3000 （補間軸は同値）
位置ループゲイン	安定な範囲で大きく	Pn102 ＝ 25 ～ 100 （補間軸は同値）
速度ループゲイン	安定な範囲で大きく	Pn100=40 ～ 80
速度ループ積分時定数	安定な範囲で小さく	Pn101=1000 ～ 2000
オートチューニング	オートチューニングしない	Pn110= ロロロ 2
位置制御による運転	位置制御を選択	Pn000= ロロ 1 ロ
位置指定フィルタ選択	加減速フィルタ	Pn207= ロロロ 0

8・2　制御操作盤参考外形図





8・3　推奨 AC サーボモータアセンブル例

	モータ形式	駆動ユニット	定　格 出　力	定格トルク (kgcm)	最大回転数 (rpm)	設定可能なモータ１回転当たりの移動量	メーカ
小型サーボ	65BM002HXE00 65BM003HXE00 65BM007HXE00 65BM010HXE00 65BM014HXE00 65BM020HXE00 65BM030HXE00	65BA008DXK00 65BA008DXK01 65BA015DXK00 65BA015DXK01 65BA030DXT00 65BA030DXT01 65BA030DXT02	50W 100W 180W 300W 400W 500W 720W	1.7 3.6 7.0 10 14 20 30	3000 3000 3000 3000 3000 3000 3000	8,4,2,1.6,1,0.8, 0.5,0.4,0.32,0.2, 0.16mm	山洋８６５シリーズ
汎用サーボ	68BM025BXE00 68BM035BXE00 68BM065BXE00 68BM090BXE00 68BM140BXE00 68BM220BXE00 68BM330BXE00	68BA030DXT00 68BA030DXT01 68BA030DXT02 68BA050DXT00 68BA075DFT00 68BA100DFT00 68BA150DFT00	0.48kW 0.66kW 1.2kW 1.6kW 2.5kW 3.6kW 4.5kW	25 35 65 90 140 220 330	2000 2000 2000 2000 2000 2000 2000		山洋８６８シリーズ
精密サーボ	USAMED-03MA1 USAMED-06MA1 USAMED-09MA2 USAMED-12MA2 USAMED-20MA2 USAMED-30MA2 USAMED-44MA2 USAMED-60MA2	CACR-PR03BB3AM CACR-PR07BB3AM CACR-PR10BB3AM CACR-PR15BB3AM CACR-PR20BB3AM CACR-PR30BB3AM CACR-PR44BB3AM CACR-PR60BB3AM	0.3kW 0.6kW 0.9kW 1.2kW 2.0kW 3.0kW 4.0kW 6.0kW	29 58 88 117 195 290 428 584	2000 2000 2000 2000 2000 2000 1500 1500	6，4，3，2.4，2，1.5， 1.2，1，0.8，0.75，0.6， 0.5，0.48，0.4，0.375， 0.3，0.25，0.24，0.2mm 上記の１倍，２倍，４倍 が設定可能。	安川Ｍシリーズ
汎用サーボ	HA-FE053 HA-FE13 HA-FE23 HA-FE33 HA-FE43 HA-FE63 HA-SE102 HA-SE152 HA-SE202 HA-SE352	MR-J10A MR-J10A MR-J20A MR-J40A MR-J40A MR-J60A MR-J100A MR-J200A MR-J200A MR-J350A	0.05kW 0.1kW 0.2kW 0.3kW 0.4kW 0.6kW 1.0kW 1.5kW 2.0kW 3.0kW	1.62 3.25 6.5 9.7 13.0 19.3 48.7 73.1 97.4 170	4000 4000 4000 4000 4000 4000 2000 2000 2000 2000	◆モータ１回転当たりの 移動量× A/B4mmA, B1 ～ 9999 1/50 ＜ /B20 電子ギア内蔵のため、 上記の範囲で設定可能。 ◆スムージング機能内蔵	三菱Ｊシリーズ
精密サーボ	HA-SA22 HA-SA52 HA-SA102 HA-SA152 HA-SA202 HA-SA352 HA-SA502 HA-SA702 HA-SA33 HA-SA53 HA-SA103 HA-SA153	MR-SA22 MR-SA52 MR-SA102 MR-SA152 MR-SA202 MR-SA352 MR-SA502 MR-SA702 MR-SA33 MR-SA53 MR-SA103 MR-SA153	0.2kW 0.5kW 1.0kW 1.5kW 2.0kW 3.5kW 5.0kW 7.0kW 0.3kW 0.5kW 1.0kW 1.5kW	9.8 24.4 48.7 73.1 97.4 170 244 340 9.8 16.2 32.5 48.7	2300 2300 2300 2300 2300 2300 2300 2300 3450 3450 3450 3450	◆モータ１回転当たりの 移動量 _A/B12mm A,B1 ～ 9999 1/50 ＜ /B50 電子ギア内蔵のため上 記の範囲で設定可能。	三菱Ａシリーズ

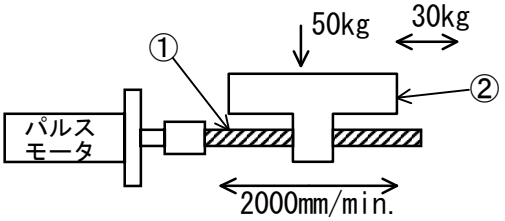
	モータ形式	駆動ユニット	定　格 出　力	定格トルク (kgcm)	最大回転数 (rpm)	設定可能なモータ 1 回転当たりの移動量	メーカ
小型サーボ	SGM-A3A31 □ SGM-A5A31 □ SGM-01A31 □ SGM-02A31 □ SGM-04A31 □ SGM-08A31 □ SGM-A3B31 □ SGM-A5B31 □ SGM-01B31 □ SGM-02B31 □	SGDA-A3AP SGDA-A5AP SGDA-01AP SGDA-02AP SGDA-04AP SGDA-08AP SGDA-A3BP SGDA-A5BP SGDA-01BP SGDA-02BP	0.03 0.05 0.1 0.2 0.4 0.8 0.03 0.05 0.1 0.2	0.974 1.62 3.25 6.49 13.0 24.3 0.974 1.62 3.25 6.49	4500 4500 4500 4500 4500 4500 4500 4500 4500 4500	◆モータ 1 回転当たりの移動量 _(Cn-24) ／ (Cn-25) ＝ 8.192mm 0.01 ≦ (Cn-24)) ／ (Cn-25) ≦ 100 電子ギア内蔵のため、上記の 範囲で設定可能。 ◆スムージング機能内蔵	安川電機Σシリーズ
精密サーボ	HA-SH52 □ HA-SH102 □ HA-SH152 □ HA-SH202 □ HA-SH352 □ HA-SH502 □ HA-SH702 □ HA-SH121 □ HA-SH201 □ HA-SH301 □	MR-H60A MR-H100A MR-H200A MR-H200A MR-H350A MR-H500A MR-H700A MR-H200A MR-H200A MR-H300A	0.5 1.0 1.5 2.0 3.5 5.0 7.0 1.2 2.0 3.0	24.4 48.7 73.1 97.4 170.0 244.0 340.0 117.0 195.0 292.0	2500 2500 2500 2500 2500 2500 2500 1200 1200 1200	◆モータ 1 回転当たりの移動量 _A/B16.384mm A,B1 ～ 50000 1/50 ＜ /B50 電子ギア内蔵のため、上記の 範囲で設定可能。 ◆スムージング機能内蔵	三菱電機Hシリーズ Cシリーズ
サーボコンパクト	HC-PQ033 □ HC-PQ053 □ HC-PQ13 □ HC-PQ23 □	MR-C10A MR-C10A MR-C10A MR-C20A	0.03 0.05 0.1 0.2	0.97 1.62 3.2 6.5	4500 4500 4500 4500	◆モータ 1 回転当たりの移動量 _A/B4.0mm 1/50 ＜ /B20 電子ギア内蔵のため、上記の 範囲で設定可能。	
新世代サーボ	BM-0130 □ BM-0230 □ BM-0430 □ BM-0730 □ BM-1020 □ BM-1520 □ BM-2020 □ BM-3020 □ BM-4020 □	BD-01 BD-02 BD-04 BD-07 BD-10 BD-15 BD-20 BD-30 BD-40	0.1 0.2 0.4 0.75 1.0 1.5 2.0 3.0 4.0	3.24 6.5 13.0 24.3 48.7 73.1 97.5 146.0 195.0	3000 3000 3000 3000 2000 2000 2000 2000 2000	◆モータ 1 回転当たりの移動量 _P/Q ＝ 10.0mm P,Q1 ～ 9999 1/50 ＜ /Q50 電子ギア内蔵のため、上記の 範囲で設定可能。	神鋼ビバップシリーズ

8・4　サーボモータ，パルスモータ選定条件

8・4・1　パルスモータ(ステッピングモータ)概略選定の手引

可動部を直線方向に移動させる場合、例題として

可動部重量＝ 50kg
推力　　＝ 30kg
移動速度　＝ 2000mm/min.
摩擦係数　＝ 0.2
伝達効率　＝ 0.9
設定単位　＝ 0.01mm



上記の条件でパルスモータのトルク，ネジのピッチ，減速比などを選定する。

1. まず、モータ 1 回転当たりの移動量として設定可能な値をリストアップする。
※モータ 1 回転当たりの移動量 ＝（モータ 1 回転当たりのステップ数）×（0.01mm）
パルスモータの基本ステップ数には 0.36゜，0.72゜，0.9゜，1.8゜があり、ドライバーによりハーフステップが可能な場合、更に 0.18゜，0.36゜，0.45゜，0.9゜が可能です。

従って、モータ 1 回転当たりの移動量として

基本ステップ 0.36°の時 ····· 10mm（ハーフステップの時 20mm）
〃　　0.72°の時 ····· 5mm（　　〃　　10mm）
〃　　0.9°の時 ····· 4mm（　　〃　　8mm）
〃　　1.8°の時 ····· 2mm（　　〃　　4mm）

上記の値が設定できます。

2. 次に、移動速度の仕様を満足させるために、
※¶ 移動速度 ＝ パルス速度 × 60 × 0.01mm
上記式より、パルス速度は 3400PPS 以上の応答周波数が必要となります。

3. 次に、1 および 2 と設計可能なネジのピッチおよび減速比からモータ 1 回転当たりの移動量を決定します。
設計例として、ネジのピッチ＝ 5mm，減速比＝ 1/1 に決定した場合、モータ 1 回転当たりの移動量は 5mm となります。
4. 次に、所要走行トルクを計算しますと、

※ 所要走行トルク　$= \dfrac{(\text{摩擦係数}) \times (\text{可動部重量}) + \text{推力}}{2\pi \times \text{効率}} \times \dfrac{\text{モータ 1 回転当たりの移動量}}{10}$

$$= \frac{0.2 \times 50\text{kgf} + 30\text{kgf}}{2\pi \times 0.9} \times \frac{5\text{mm}}{10} = 3.6\text{kgf·cm}$$

5. 各メーカー発行のカタログよりパルスモータの速度トルク特性を参照して、3400PPS で 3.6kgf・cm 以上の特性をもつタイプを検討します。

6. 前記の結果より安全率を考慮して選定いたしますと、オリエンタルモータ 5 相ステッピングモータ UPD569 タイプの場合、4000PPS で 約 7kgf・cm のトルクがありますので、上記仕様に適していることがわかります。

7. 次に負荷の慣性モーメントを計算いたしますと、ボールネジの径を Φ15mm，長さ 600mm とした場合、

①部の慣性モーメント＝ $\pi/32 \times \rho \times L \times D^4$ ： ρ ＝ 鉄の比重 ＝ 7.9×10^{-3}kg/cm^2

＝ $\pi/32 \times 7.9 \times 10^{-3} \times 60 \times (1.5)^4 = 0.235$kg・cm^2

②部の慣性モーメント＝ W（P/2π）^{2} ：P ＝ ボールネジのピッチ ＝ 5mm

＝ 50（0.5/2π）^{2} ＝ 0.317kg・cm^2

モータ軸換算の負荷慣性モーメントは直結ですから、減速比を 1/1 にすること

負荷慣性 ＝ 0.235（1/1）^{2} ＋ 0.317（1/1）^{2} ＝ 0.552kg・cm^2

モータの慣性はカタログより 0.4kg・cm^2，自起動周波数は 1000PPS ですから

$$※ \text{自起動周波数} = \frac{\text{モータ単体の自起動周波数}}{\sqrt{1 + \dfrac{\text{負荷の慣性}}{\text{モータの慣性}}}} \times \frac{1000}{1.54} = 649\text{PPS}$$

となり、自起動周波数以上の 3400PPS で動作させるためには、入力パルスをスローアップ，ダウンさせる必要があります。

8. 負荷の慣性がモータの慣性の 2 倍以上ある場合や、自動周波数以上の高速回転が必要な場合は、プログラムで速度を徐々に上げるか、または加減速を入れることでパルスモータの脱調を防止することが可能です。
この場合の加減速カーブは指数形となり、理想的な加減速に近くなります。

9. 上記の他、パルスモータやドライバーの運転条件，振動，温度上昇など、詳細はメーカー発行の技術資料を御参照ください。

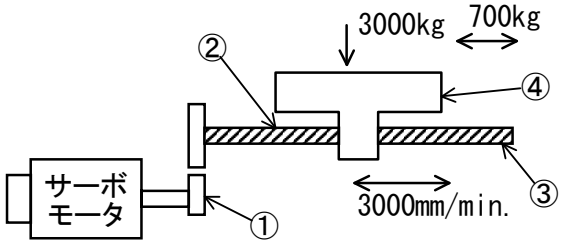
サーボ系を設計するに際し重要な点は、可能な限りモータ軸換算の負荷慣性を小さくすることです。負荷慣性が大きい場合は加減速時間を長くしなければいけないため、高速円弧切削時の精度が悪くなります。

8・4・2　サーボモータ概略選定の手引

(A) 直線運動の場合

例題として

可動部重量　＝ 3000kg
切削版力　　＝ 700kg
設定単位　　＝ 0.001mm
送り速度　　＝ 3000mm/min.
摩擦係数　　＝ 0.2
伝達効率　　＝ 0.9



1. まず、モータ１回転当たりの移動量として設定可能な値をリストアップする。

※モータ１回転当たりの移動量＝（モータ１回転当たりのフィードバックパルス数）×（0.001mm）

安川Ｍシリーズ AC サーボの場合、モータ１回転当たりのフィードバックパルス数は

$6000 \times \frac{1}{N}$（N＝1,2,3･･･32）　　または　$6000 \times \frac{2}{N}$（N＝2,3,4･･･32）

$5000 \times \frac{1}{N}$（N＝1,2,3･･･32）　　または　$5000 \times \frac{2}{N}$（N＝2,3,4･･･32）

上記式で計算され、モータ１回転当たりの移動量として、

検出器 6000P の時　6，4，3，2.4，2，1.5，1.2，1，0.8，0.75，0.6，0.5，0.48，0.4，0.375，0.3，0.25，0.24，0.2mm

検出器 5000P の時　5，2.5，1.25，0.625mm

上記の１，２，４倍が設定できます。

2. 次に、送り速度の仕様を満足させるために、
※送り速度 Max. ＝（モータ定格回転数）×（モータ１回転当たりの移動量）

安川Ｍシリーズ AC サーボの場合、モータ定格回転数は 1000rpm ですから、モータ１回転当たりの移動量は 3mm 以上が必要となります。

3. 次に、1 および 2 と設計可能なネジのピッチ，減速比からモータ１回転当たりの移動量を決定します。
設計例として、ネジのピッチ＝ 6mm，減速比＝ 1/2 に決定した場合、モータ１回転当たりの移動量は 3mm となります。

4. 次に、所要走行トルクを計算します。

$$※所要走行トルク = \frac{(摩擦係数)\times(可動部重量)+切削反力}{2\pi \times 効率} \times \frac{モータ1回転当たりの移動量}{10}$$

$$= \frac{0.2\times 3000kgf+700kgf}{2\pi \times 0.9} \times \frac{3mm}{10} = 69kgf\cdot cm$$

5. 次に、モータ軸換算の負荷慣性を計算し、許容値内にあるか検討します。

①部のタイミングベルトのプーリーを φ90, 巾 24 mm の中実円筒にすると、
$\pi/32 \times \rho \times L \times D^4 = \pi/32 \times 7.9 \times 10^{-3} \times 2.4 \times (9)^4 = 12.2kg\cdot cm^2$

②部のプーリーは、φ180, 巾 24 mm の中実円筒として、
$\pi/32 \times 7.9 \times 10^{-3} \times 2.4 \times (18)^4 = 195.3kg\cdot cm^2$　　減速比は 1/2 だから
モータ軸換算の負荷慣性は　$195.3 \times (1/2)^2 = 48.8kg\cdot cm^2$

③部のボールネジは φ50, 長さ 3000mm の中実円筒として、
$\pi/32 \times 7.9 \times 10^{-3} \times 300\ (5)^4 \times (1/2)^2 = 36.3kg\cdot cm^2$

④部のテーブルは重量 3000kg, ボールネジのピッチ 6mm として、
$W \times (P/2\pi)^2 = 3000 \times (0.6/2\pi)^2 = 27.4kg\cdot cm^2$　　減速比は 1/2 だから
モータ換算の慣性は　$27.4 \times (1/2)^2 = 6.85kg\cdot cm^2$

以上の①②③④部のモータ軸換算負荷慣性の合計は、
$J = GD^2/4 = 12.2 + 48.8 + 36.3 + 6.85 = 104.15kg\cdot cm^2$
【注】J（kg・cm^2）= GD^2/4（kg・cm^2）= J（g・cm・S^2）× 0.98

6. モータのカタログより、定格トルク 69kgf・cm 以上,
モータ軸換算の負荷慣性≦モータのロータ慣性 ×2 の条件を考慮して選択いたしますと、
モータ ..... USAMED-12MA2　（6000P 検出器仕様）
サーボパック ..... CACR-PR15BB3AM （6000P 検出器仕様）

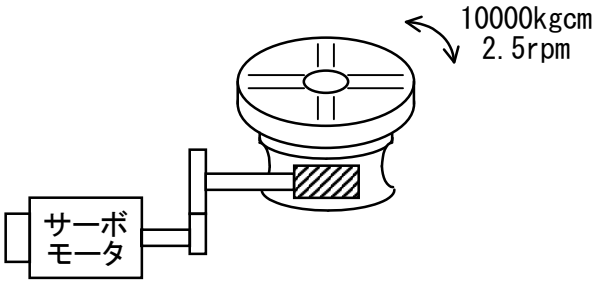
上記が 定格トルク 117kgf・cm, GD^2/4 = 66.8kg・cm^2 で条件を満足しますから、推奨されます。
(正確な判断データが必要な場合は次項の所要最大トルクの計算も行ってください。)
三菱電機の場合、上記相当のモータおよびアンプを選定いたしますと、
モータ HA-SE202
サーボパック MR-J200A
設定可能な 1 回転当たりの移動量は電子ギア機能がありますから、
モータ 1 回転当たりの移動量（3mm）×A/B = 4mm の式から A = 4, B = 3 で設定可能です。

サーボ系を設計するに際し重要な点は、可能な限りモータ軸換算の負荷慣性を小さくすることです。負荷の慣性が大きい場合は、位置ループゲインを上げると振動が発生し、補間制御に必要な高ゲイン（KP = 25 以上）に設定できないため、切削時の精度が悪くなります。

(B) 回転運動の場合

例題として

負荷トルク＝ 10000kgcm
回転速度　＝ 900deg./min. ＝ 2.5rpm
伝達効率　＝ 0.7
設定単位　＝ 0.001deg.



1. まず、モータ 1 回転当たりの回転角として設定可能な値をリストアップする。

※モータ 1 回転当たりの移動量＝（モータ 1 回転当たりのフィードバックパルス数）×（0.001deg.）

安川 M シリーズ AC サーボの場合、モータ 1 回転当たりのフィードバックパルス数は、直線運動の場合と同じですから、

検出器 6000P の時　6，4，3，2.4，2，1.5，1.2，1，0.8，0.75，0.6，0.5，0.48，0.4，0.375，0.3，0.25，0.24，0.2deg.

検出器 5000P の時　5，2.5，1.25，0.625deg.

上記の 1，2，4 倍が設定できます。

2. 次に、送り速度の仕様を満足させるために、
※送り速度 Max. ＝（モータ定格回転数）×（モータ 1 回転当たりの回転角）
安川 M シリーズ AC サーボの場合、モータ定格回転数は 1000rpm ですから、モータ 1 回転当たりの回転角は 0.9deg. 以上が必要となります。

3. 次に、1. および 2. と設計可能な減速比から、モータ 1 回転当たりの回転角を決定します。
設計例として 減速比＝ 1/360 に決定した場合、モータ 1 回転当たりのテーブル回転角は 1deg. となります。

4. 次に、所要走行トルクを計算します。

$$※\ \text{所要回転トルク} = \frac{\text{負荷トルク}}{\text{効率}} \times \frac{\text{モータ 1 回転当たりの回転角}}{360}$$

$$= \frac{10000\text{kgf·cm}}{0.7} \times \frac{1}{360} = 39.7\text{kgf·cm}$$

5. 次に、モータ軸換算の負荷 GD^2 を計算し、所要最大トルクを計算します。

$$※\ 所要最大トルク\ =\ \frac{(モータGD^2+負荷GD^2)\ \times モータ回転数}{375\times 加速時間}\ +\ 所要回転トルク$$

仮にサーボモータを安川の AC サーボ 09MA とし、負荷 GD^2 がモータ GD^2 の２倍の場合、カタログより $GD^2/4 = 36.7kg\cdot cm^2$　　故に　モータ $GD^2 = 146.8\times 10^{-4}\ kg\cdot cm^2$
位置ループゲインを 30 にし、25mS の加減速を行った場合、
加速時間 ＝ 1.4/ 位置ループゲイン ＋ 1.4 × 加速時定数
　　＝ 1.4/30 ＋ 1.4 × 0.025 ＝ 0.082sec.

$$\frac{(146.8\ \times\ 10^{-4}\ +\ 293.6\ \times\ 10^{-4})\ \times\ 900}{375\ \times\ 0.082}\ =\ 1.3kgf\cdot m\ =\ 130kgf\cdot cm$$

所要最大トルク ＝ 130kgf・cm ＋ 39.7kgf・cm ＝ 169.7kgf・cm となります。
これらの値はモータの最大トルク（197kgf・cm），定格トルク（88kgf・cm）を満足しますから、09MA が使用可能となります。

6. 上記の結果により安全率などを考慮して選定いたしますと、

モータ .....USAMED-09MA2 （6000P 検出器仕様）
サーボパック.....CACR-PR10BB3AM（6000P 検出器仕様）
が推奨されます。

7. その他の注意事項

a）負荷の慣性はモータ慣性の３倍以内になるように伝導系を設計してください。
　モータ軸換算の負荷慣性≦モータの慣性 × 3（推奨値は２倍以内）
　（モータ軸換算の負荷慣性が大きいほどサーボ性能は悪くなります。）

b）上下方向の軸に対しては、バランサーおよびブレーキを考慮してください。
バランサーが無い場合、重力によるトルクはモータの定格トルクの 1/2 以下としてください。

c）サーボゲインは負荷に適した値で、なるべく高く（標準位置ループゲイン 25 ～ 40）調整してください。　また、補間制御をする軸の位置ループゲインは同じ値に設定ください。

d）加減速時間を長く設定しますと、円弧切削を高速で行う場合の精度が悪くなります。

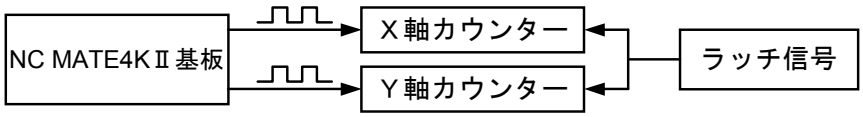
e）三菱電機 J,H シリーズ、安川電機 Σ シリーズには電子ギア機能がありますので、設定可能なモータ１回転当たりの移動量は、電子ギアにより補正が可能です。

f）詳細につきましては、サーボモータメーカーの技術資料を参照してください。

8・5　円弧補間の精度測定データ

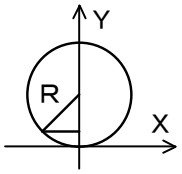
この資料は基板よりの出力パルスをカウンターにて計測し、補間時の精度を測定したデータです。

〔測 定 方 法〕



〔測 定 条 件〕　Ｒ＝10mm　　Ｆ＝100～2000mm/min.　　スキャンタイミング＝10.24mS

〔プログラム〕　G17 G02 X10 Y10 R-10 F1000
G02 X-10 Y-10 R10



移動速度	Ｘ軸測定値	Ｙ軸測定値	$\sqrt{X^2+(10-Y)^2}$	誤　差　Δｒ
100mm/min.	−5.357 −8.612 8.606 4.445	1.556 15.082 15.093 1.042	9.9999292 9.9996633 10.000093 10.000188	−0.071μ −0.337μ +0.093μ +0.188μ
500mm/min.	−4.889 −8.002 8.458 5.221	1.277 15.997 15.335 1.471	9.9996524 9.9998006 9.9999994 10.000133	−0.348μ −0.2μ −0.001μ +0.133μ
1000mm/min.	−4.742 −5.622 8.911 4.518	1.196 18.270 14.538 1.079	9.9998489 9.9999891 9.9999682 9.9998282	−0.152μ −0.011μ −0.032μ −0.172μ
2000mm/min.	−6.950 −3.688 6.720 4.534	2.811 19.294 17.404 1.088	9.9992110 9.9989889 9.9988807 9.9990449	−0.789μ −1.012μ −1.12μ −0.956μ

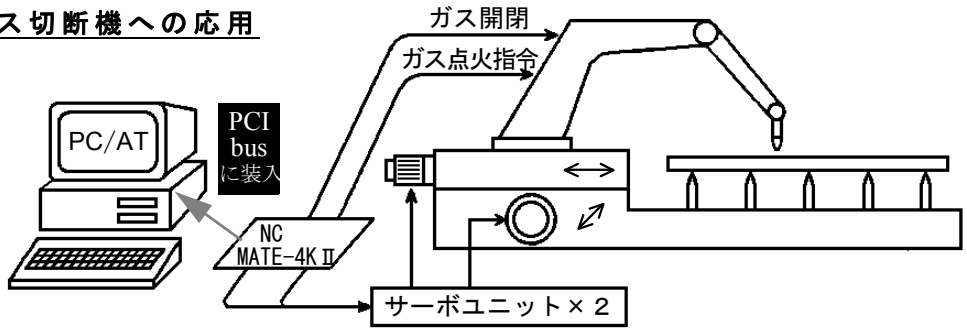
【考察】上記データより、誤差は１μþ程度で最小設定単位の量子誤差以内であり、サーボモータに起因する誤差に比較して十分小さいので問題ありませんが、更に補間精度を向上させるためにはスキャンタイミングを短く設定することにより精度向上が可能です。

【備考】円弧補間の精度を向上させるためには、補間軸の位置ループゲインを合わせると共に、できるだけ高く設定する必要があります。
また、サーボのゲインを高くするためには、負荷の慣性が小さくなるように伝導系を設計する必要があります。

8・6　NC MATE-4K Ⅱの応用例

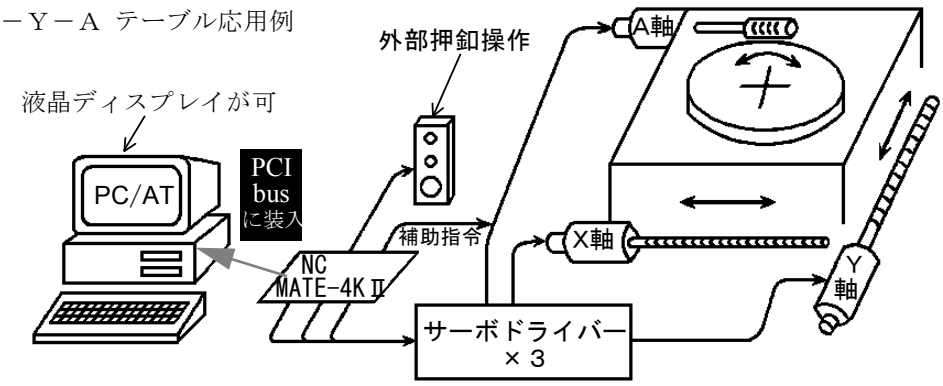
本製品は、様々な用途でご使用いただけます。

１．ガス切断機への応用

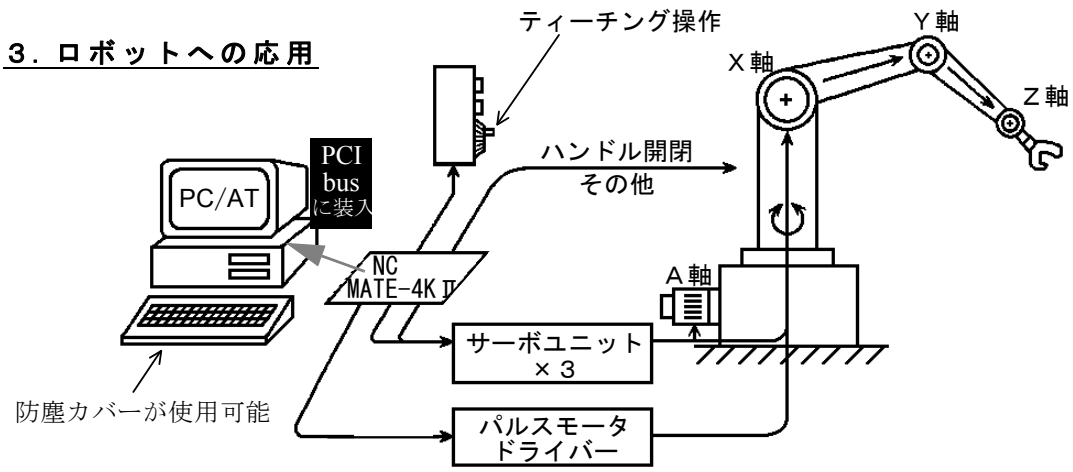


２．数値制御の教材

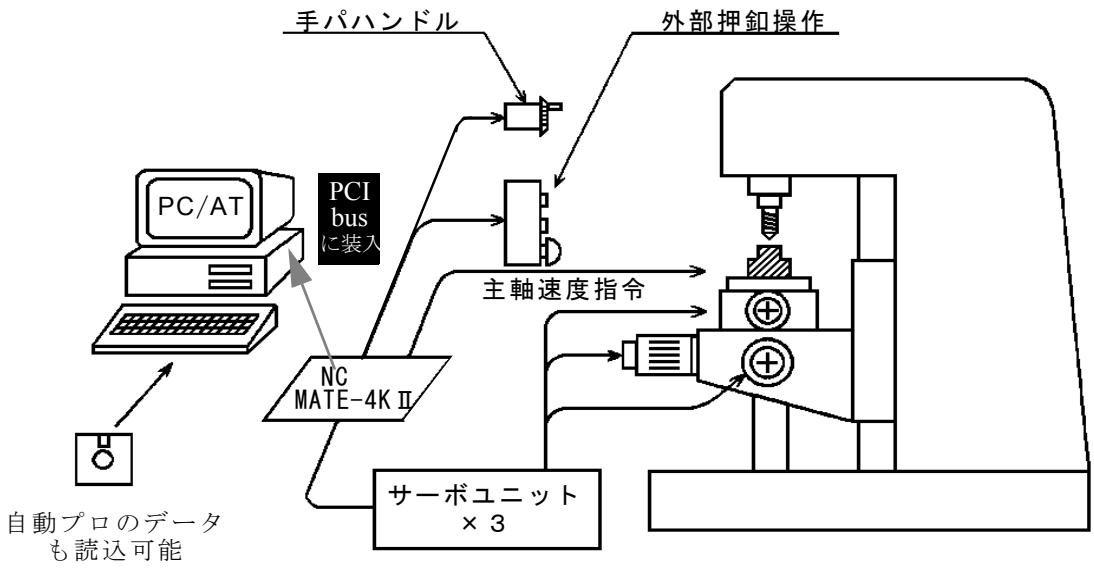
Ｘ－Ｙ－Ａ テーブル応用例



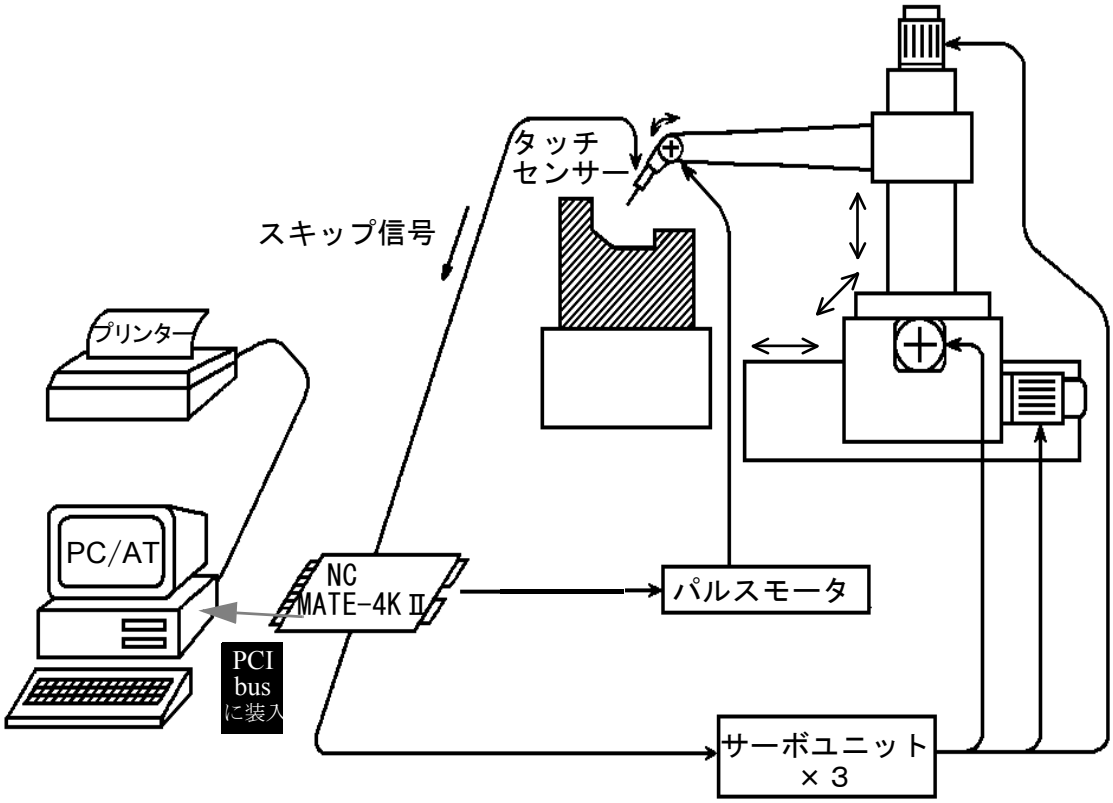
３．ロボットへの応用



４．工作機械への応用



５．計測装置の応用



６．その他の応用例

機種	用途
ミーリング	一般金属加工 （鋳物，鋼材）
	非鉄，セラミック加工
	特殊加工 （タービンブレード加工）
研削盤	一般研削 （平面研削，円筒研削）
	ボールネジ研削
	ウォーム研削
	大型ロール研削 （圧延ロール）
	セラミック研削 （ターボファン）
	歯車研削
ルータ	NC ルータ
	プラスチック，プリント基板加工
	マーク，銘板彫刻
切断機	プラズマ切断機，ガラス切断機
	レーザー加工機
Ｘ－Ｙテーブル	工業用ミシン
	写真焼付光学装置
成形機	セラミックス成形機
	金属成形機
歯車加工機	ギヤシェーパー
穴加工機	ボール盤，タッピング加工
マーキング装置	マーキング （金属，プラスチックなど）
ロボット	搬送ロボット，バリ取りロボット
ＩＣ関係	シリコンウェハー面取り加工
計測装置	デジタイザー，寸法計測装置
実験装置	プロペラ流体実験，摩擦係数実験装置
試験装置	繰り返し位置決め試験，耐久テスト装置
教育関係	NC 制御の実習教育，プログラム作成練習

最新情報は、ホームページでご案内しております。各種製品情報、マニュアル等のダウンロード、サポートのご案内など、さまざまな情報が満載です。

Kyopal ホームページ ▶ http://www.kyopal.co.jp/

製品に関する技術的なお問い合わせは…	製品のサンプル貸出についてのお問い合わせは…
support@kyopal.co.jp	sales@kyopal.co.jp

取扱店

制御ボード&オリジナルLSIで、FA要素技術を提案する

株式会社キョーパル
京都市下京区西七条東石ヶ坪町24 〒600-8895
TEL(075)326-2580(代) / FAX(075)326-2581

●製品改良のため、予告なく仕様を変更する事がありますので、ご了承ください。08 JUN